

Cyclic Perturbation: Protecting Confidentiality While Preserving Data Utility in Tabular Data

George T. Duncan*

Stephen F. Roehrig †

H. John Heinz III School of Public Policy and Management

Carnegie Mellon University

Pittsburgh, PA, USA

June 3, 2004

Keywords: Statistical Disclosure Limitation, Confidentiality, Cell Suppression, Rounding

Abstract

When disseminating data on individuals, information organizations must balance the interests of data users in better access and the interests of data providers in confidentiality. *Cyclic perturbation* is a new method for protecting sensitive data in categorical tables. In the disseminated data product, the true table values are altered in a way that preserves the table's marginal totals. Further, the method requires publicly announcing details of the procedure, so that a user can determine not only the range of values that a table cell may have had in the original table, but also the *exact* posterior distribution over those possible values, given the user's prior probabilities over a relatively small set of possible true tables. This permits Bayesian analysis of the published table. Cyclic perturbation is compared with the standard alternatives of cell suppression and controlled rounding, on the basis of disclosure risk and data utility.

1 Introduction

Consistent with their mandate, statistical agencies and other information organizations (IOs) must resolve the tension between the demands of data users for better access and the demands of data providers for privacy and confidentiality (Duncan, Jabine and de Wolf 1993, Willenborg and De Waal 1996). Although

*Supported by NSF under Grants EIA-9876619 and EIA 0131884, NCES under Agreement EDOERI-00-000236, NIA under Grant 1R03AG19820-81

†Supported by NSF under Grant EIA-0131884 through the National Institute of Statistical Sciences

agencies sometimes release microdata products after disclosure limitation, the most common mode of dissemination is through tables (Duncan, Fienberg, Krishnan, Padman and Roehrig 2001). Typically these tables have been in printed form, but increasingly they are accessed through the web, e.g, the U.S. Census Bureau’s American FactFinder (<http://factfinder.census.gov/>, Hawala, Zayatz and Rowland 2004).

Recognizing that release of data products entails disclosure risk, IOs typically employ some form of disclosure limitation. Most often with tabular data, IOs have employed cell suppression (Cox 1980, Cox 1995, Kelly, Golden and Assad 1992) to lower disclosure risk. Under cell suppression the values of table cells that pose confidentiality problems are determined and suppressed (as primary suppressions) as well as values of additional cells that can be inferred from released table margins (as secondary suppressions). Cell suppression has critical deficiencies for practical use, however. In particular, the “all-or-nothing” nature of cell suppression can lead to a “Sophie’s Choice” between dissemination resulting in high disclosure risk or dissemination with suppression resulting in low data utility. Also, determination of appropriate secondary suppressions is computationally difficult for multi-way tables (Kelly, Golden and Assad 1992). Other disclosure limitation techniques that have been used include global recoding and various forms of perturbation (Subcommittee on Disclosure Limitation Methodology 1994). Perturbation is used through controlled rounding (Cox 1987), versions of post-randomized response (de Wolf, Gouweleeuw, Kooiman and Willenborg 1998) and Markov perturbation approaches (Duncan and Fienberg 1998, Fienberg, Makov and Steel 1998). Many of these methods can be represented in the form of matrix masks (Duncan and Pearson 1991). For practical implementation, information managers want to be able to compare these procedures in terms of their disclosure risk R and data utility U , and specifically explore tradeoffs between R and U . This can be done through the R - U confidentiality map, as demonstrated in (Duncan, Fienberg, Krishnan, Padman and Roehrig 2001). In this article, we introduce a promising new disclosure limitation method that we call cyclic perturbation, and investigate its properties relative to other methods using the R - U confidentiality map.

The basic confidentiality problem is protecting against a data snooper attack by lowering disclosure risk while providing value to legitimate data users by maintaining data utility (Section 2). We introduce cyclic perturbation (Section 3) as a new disclosure limitation method for tabular data. An example is developed (Section 4) that shows how the method can be applied, and leads to tractable analysis in a Bayesian framework. In general, we show that with a uniform prior distribution over the possible tables that may have led to the released table, the posterior mode for a cell value coincides with the published cell value (Section 5). The implications of specifying prior distributions (Section 6) are developed, as well as appropriate specifications for data utility and disclosure risk (Section 7). Cyclic perturbation is compared (Section 8) with other disclosure limitation methods.

Individual	v	w
1	v_1	w_3
2	v_1	w_2
3	v_4	w_3
4	v_2	w_1
5	v_1	w_3
6	v_3	w_4
\vdots	\vdots	\vdots

Table 1: Typical display of microdata

	w_1	w_2	w_3	w_4	
v_1	15	1	3	1	20
v_2	20	10	10	15	55
v_3	3	10	10	2	25
v_4	12	14	7	2	35
	50	35	30	20	135

Table 2: A typical cross-classification

2 Disclosure Risk and Data Utility

Consider a database containing individual identifiers and the values of two attributes V and W , collected for each of n individuals. (Our remarks, and methods, apply to higher-dimensional databases and tables as well.) If we denote the possible values of the first attribute $v_i, i = 1, \dots, I$, and those of the second $w_j, j = 1, \dots, J$, the data can be presented as a list of *microdata* records, as in Table 1.

A tabular representation of these data is a two-dimensional table of non-negative integers $T = [t_{ij}]$, containing in each cell the count of individuals possessing identical values for the two attributes. Table 2 gives a typical cross-classification (Cox, McDonald and Nelson 1986, Subcommittee on Disclosure Limitation Methodology 1994).

An interior cell in a table like this is typically considered sensitive if its value is small, say 1 or 2, because knowledge of its exact value could reveal confidential information about an individual respondent. Our task is to mask these sensitive cells in order to preserve confidentiality. In this example there are four cells with sensitive values: $((v_1, w_2), (v_1, w_4), (v_3, w_4), (v_4, w_4))$. A second goal, beyond protecting confidentiality, is to retain in the masked table as much statistical information as possible, so that the table as a whole is useful to data users.

Many methods for achieving these dual goals of confidentiality protection and data utility for tabular data have been proposed (Cox 1995, Cox 1987, Fischetti

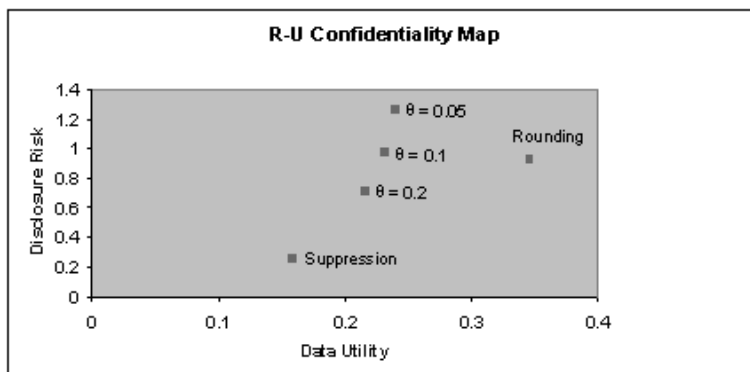


Figure 1: An R-U confidentiality map (from Duncan, Fienberg, Krishnan, Padman and Roehrig 2001)

and Salazar 1998, Gopal, Goes and Garfinkel 1998, Gusfield 1988, Kelly, Golden and Assad 1992, Kelly, Golden, Assad and Baker 1990, Willenborg and De Waal 1996). Our aim is to first provide a framework for assessing the success of such methods, and then advance a new method, called cyclic perturbation. After examining cyclic perturbation using the framework provided, we compare it with the well-known methods of cell suppression and controlled rounding.

The framework we use in addressing the twin issues of disclosure risk and data utility is called an *R-U* confidentiality map (Duncan and Fienberg 1998, Duncan, Fienberg, Krishnan, Padman and Roehrig 2001). In its simplest form, it is a set of (R, U) pairs of disclosure risk and data utility, that quantify different release strategies available to the IO. These strategies may be distinct disclosure limitation techniques, for example cell suppression vs. controlled rounding, or may comprise a range of parameter values governing a fixed disclosure limitation procedure, as in the variance associated with a noise addition process or the rounding base used in controlled rounding. Visually, the (R, U) pairs in the map trace the tradeoffs available to the IO. The horizontal axis in the map indicates data utility, the usefulness of a data release to its intended users. The vertical axis measures disclosure risk, the degree to which the data release compromises sensitive information.

One example of an *R-U* confidentiality map appears in (Duncan, Fienberg, Krishnan, Padman and Roehrig 2001), and is reproduced here as Figure 1. In this figure, three disclosure limitation methods are compared for a two-way table: cell suppression, controlled rounding to base 3, and Markov perturbation, for three values of the perturbation parameter θ . The measure of disclosure risk is the entropy of the probability distribution for values of a fixed sensitive cell, defined as $1/(-\sum(p_\omega \log p_\omega))$, where ω ranges over the sensitive cell's possible values. Data utility is measured as mean squared precision, the reciprocal of the mean squared error of the distribution for ω , given its true value.

The *R-U* confidentiality map allows the determination of a Pareto-optimal

frontier and an examination of tradeoffs between disclosure risk and data utility on this frontier. In some cases, this suggests selecting an optimal disclosure limitation method by first determining a maximum tolerable disclosure risk, and then by choosing the method with the greatest data utility. Having the R - U confidentiality map as a tool to simultaneously examine disclosure risk and data utility, we can compare various disclosure limitation procedures. In the next section we present a new technique for disclosure limitation of tabular data, called cyclic perturbation.

3 Cyclic Perturbation

The fundamental idea behind cyclic perturbation builds on earlier work (Duncan and Fienberg 1998, de Wolf, Gouweleeuw, Kooiman and Willenborg 1998). We alter the true table values in a way that preserves the table’s marginal totals, and then publish the altered table. Thus the method we describe is related to controlled rounding (Cox 1987, Kelly, Golden, Assad and Baker 1990), and has connections to cell suppression as well. What is new is the flexible and intuitive stochastic procedure used to produce the altered table. The procedure allows a variety of perturbation patterns and has useful analogies to misclassification in frequency tables, just as additive noise has useful analogies to measurement error for continuous microdata. Further, cyclic perturbation, by requiring public announcement of details of the procedure, allows a user to determine not only the range of values that a table cell may have had in the original table, but also the *exact* posterior distribution over those possible values, given the user’s prior probabilities over a relatively small set of possible true tables. This permits a Bayesian analysis of the published table, and the determination of an R - U confidentiality map.

Our approach has distinct advantages over existing methods because the posterior cell distributions can be computed by both the data disseminator and data users. The disseminator has clear knowledge of the level of protection afforded the table, and users can take these distributions into account when analyzing the published table. Other disclosure limitation methods for tables do not provide sufficient information to compute these distributions. We discuss this further in Section 8.

To protect the interior cells of a table, we modify their values in a principled way, by applying a sequence of “perturbations” to them. These perturbations leave the marginal totals unchanged. Each perturbation modifies a patterned collection of four or more cells—called a *data cycle*—with some cell values increasing by one and others decreasing by one, in such a way that each row and column sum is undisturbed.

A data cycle can be depicted as a table containing cells marked “+”, “0” and “−”. The simplest data cycles are *elementary data squares*. These consist of two “+” signs and two “−” signs arranged in a square, with all other cells containing “0”. Three example data cycles for a 4×4 table, the first two of length 4 (i.e., data squares) and the third (of length 6), are the following.

+	0	-	0
-	0	+	0
0	0	0	0
0	0	0	0

0	0	+	-
0	0	0	0
0	0	-	+
0	0	0	0

+	0	0	-
-	0	+	0
0	0	-	+
0	0	0	0

There are $\binom{m}{2} \binom{n}{2}$ elementary data squares for an $m \times n$ table, and it is easy to see that any more complicated data cycle can be constructed by forming sums and differences of these squares. For example, the third cycle above is the sum of the first two. Note that a cyclic perturbation is a special type of matrix mask (Duncan and Pearson 1991), one involving only an additive transformation having row and column totals zero.

Once a set of data cycles has been chosen to protect the original table, cyclic perturbation proceeds by applying each cycle once, in a fixed order, to the table. At each step, we flip a three-sided coin, whose probabilities for sides A, B, and C are α , β and $\gamma \equiv 1 - (\alpha + \beta)$, respectively. If the coin shows side A, data table cells in the positions marked + in the data cycle are increased by one, and cells in positions marked - are decreased by one. If the coin shows side B, the reverse happens, while if the coin shows side C, the data table is unchanged. Each such move, and so any sequence of such moves, leaves the row and column sums unchanged.

There are two useful ways of thinking about cyclic perturbations. The first is that of interconnected random walks. This permits some well-understood probability theory to be applied. The second is that cyclic perturbation can be thought of as a process of stochastic, linked misclassifications of pairs of entities. For example, if the coin flip produces side A, the first data cycle depicted above is a data swap in which the two rows of the data corresponding to individuals 1 and 4 in Table 1 have their second (w) components exchanged. If the coin comes up side B, the perturbation exchanges these individuals' v components.

The intuition behind cyclic perturbation as a disclosure limitation procedure is that the parameters α and β can be chosen "sufficiently large" to provide adequate confidentiality protection for the interior cells. The values of α and β may be considered parameters in the disclosure limitation process, so their choice is informed by the R - U confidentiality map.

In the special case where a cycle contains a cell entry whose current value is zero, both α and β are set to zero. That is, any cell having a zero value is constrained to remain there. This choice is motivated, first, by the data swapping interpretation given above. If the original data contain no instance of a particular combination of attributes, then no swap can be performed. A second motivation, directly related to the first, is that *structural zeros*, cells that must contain a zero for logical reasons (e.g., pregnant males), should always remain zero.

If α and β are set equal, it is easy to show that after any number of perturbations, the expected value of any cell is just its original value. That is, the process of cyclic perturbation is unbiased. This is still true even with the special rule for data cycles containing zeros given above, since in this case $\alpha = \beta = 0$.

For this reason, it is desirable in practice to set $\alpha = \beta$.

If the process of cyclic perturbation is continued indefinitely, with cycles chosen arbitrarily from the set of all possible cycles, every cell in the table undergoes a random walk with two absorbing states, one at the maximum possible value and one at the minimum possible value. It is essential that the cycles not be restricted to, say, pairwise adjacent data squares. If they are, the limiting values are not necessarily the maximum and minimum possible cell values. Choosing cycles from the complete set assures that the maximum and minimum cell values can be reached. The situation is even more delicate in higher dimensions. See (Diaconis and Sturmfels 1998) for details. To illustrate this absorbing characteristic of cyclic perturbation, one sequence of moves chosen to maximize the upper left cell (v_1, w_1) results in the following table (for which we no longer show the marginal totals, since they are unchanged).

20	0	0	0
30	0	25	0
0	25	0	0
0	10	5	20

Once this state is reached, there is no data cycle containing all positive values, and so the state is absorbing. A different sequence of data cycles could also reach the following state, which also has no data cycle containing all positive values.

20	0	0	0
30	25	0	0
0	0	25	0
0	10	5	20

As this example shows, the order in which the data cycles are chosen is critical in determining which tables can be reached through cyclic perturbation. Our overall goal is to allow IOs and data users to determine the set of tables that *could* be the original, unperturbed table, as this will (as we show) permit a precise analysis of disclosure risk and data utility. Since we want to analyze these properties for both a given realization of a set of perturbations *and* for the method as a whole (say through expected values of these measures), choosing data cycles at random is unsatisfactory.

To overcome this difficulty, we make the sequence of data cycles explicit. That is, the IO protecting the data chooses, *and publishes*, the collection and order of the data cycles that are applied. The values of α and β may or may not be published. If not published, an extra level of security is provided, but this will considerably lessen the utility of the table to the data user.

Suppose a sequence of data cycles is applied to the original table T^O . The result will be a table T^P , whose marginal totals agree with those of the original table, but whose internal cells may have new values. The perturbation process is illustrated schematically in Figure 2. In this figure, T^O is the original table,

T^P is the table after two perturbations, and the operators M_1, M_2 are the perturbations. At the right of the figure is a tree representation of the process. Since for each perturbation M_i there are three possible outcomes (corresponding to A, B, or C from the coin flip), there are three arcs extending to the right of each node.

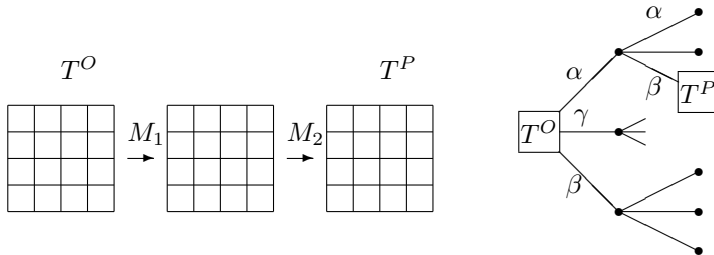


Figure 2: Cell Perturbation Process

The perturbed table T^P is a candidate for dissemination. In order to assess the disclosure risk R and the data utility U of T^P , we want to know how much the knowledge of T^P reveals about the original table T^O . As a starting point, compute for each cell (v_i, w_j) , the conditional probability $\Pr(t_{ij}^O | T^P)$ for each possible value of t_{ij}^O . We do this by first calculating $\Pr(T^P | T_k)$ for all tables T_k that could have given rise to T^P through the perturbation process (thus $T^O \in \{T_k\}$). (We discuss how $\{T\}$ can be generated in the next section.) Then we fix a prior distribution over the set of such tables $\{T_k\}$. Next, Bayes' theorem is applied to find $\Pr(T_k | T^P)$ for each $T_k \in \{T_k\}$. Finally, posterior probabilities $\Pr(t_{ij}^O = q | T^P)$ are computed by summing probabilities over those tables T_k having $t_k(ij) = q$.

For any specific outcome T^P of cyclic perturbation, this procedure allows disclosure risk to be assessed, since the distribution over the possible cell values for T^O can be calculated. For example, if we chose as the measure of disclosure risk

$$R = \max_{ij} \{\Pr(t_{ij}^O = 1 \text{ or } 2)\},$$

its value can be immediately calculated from the posterior distribution.

4 Example and Analysis in a Bayesian Framework

Consider our running example, which has

$$T^O = \begin{bmatrix} 15 & 1 & 3 & 1 \\ 20 & 10 & 10 & 15 \\ 3 & 10 & 10 & 2 \\ 12 & 14 & 7 & 2 \end{bmatrix}.$$

Choose the following sequence of data cycles:

$$M_1 = \begin{bmatrix} + & - & 0 & 0 \\ 0 & + & - & 0 \\ 0 & 0 & + & - \\ - & 0 & 0 & + \end{bmatrix} \quad M_2 = \begin{bmatrix} 0 & + & - & 0 \\ 0 & 0 & + & - \\ - & 0 & 0 & + \\ + & - & 0 & 0 \end{bmatrix}$$

$$M_3 = \begin{bmatrix} 0 & 0 & + & - \\ - & 0 & 0 & + \\ + & - & 0 & 0 \\ 0 & + & - & 0 \end{bmatrix} \quad M_4 = \begin{bmatrix} - & 0 & 0 & + \\ + & - & 0 & 0 \\ 0 & + & - & 0 \\ 0 & 0 & + & - \end{bmatrix}$$

If the coin flips come out as [A, A, B, C], we end up with the following table.

$$T^P = \begin{bmatrix} 16 & 0 & 2 & 2 \\ 21 & 11 & 9 & 14 \\ 2 & 11 & 11 & 1 \\ 11 & 13 & 8 & 3 \end{bmatrix}.$$

For this realization of T^P , only a certain set of tables T_k could have produced T^P using the given sequence of data cycles, (M_1, M_2, M_3, M_4) . T^O is of course one of the T_k . These tables are determined by constructing a tree, starting at T^P and recursively spreading “backward” using the data cycles in reverse order. This tree is built by identifying, at each stage, every table that could have been perturbed into a table identified at the preceding stage, taking into account only those that do not have a zero cell value somewhere in the current data cycle. Because of this restriction, only 45 of the $3^4 = 81$ potential predecessors are feasible. Each leaf table T_k has an associated probability $\Pr(T^P \mid T_k)$ specifying, in terms of the values of α, β , and γ , the probability of ending up with T^P given that T_k was the original table. For example, when using preorder traversal of the tree, the 4th table found is

$$T_4 = \begin{bmatrix} 16 & 1 & 1 & 2 \\ 21 & 11 & 10 & 13 \\ 1 & 11 & 11 & 2 \\ 12 & 12 & 8 & 3 \end{bmatrix},$$

with the sequence $[\alpha, \alpha, \gamma, \alpha]$ from T^P , so having T^P as a starting point would produce T^P , with probability $\alpha\alpha\gamma\alpha$. That is, $\Pr(T^P \mid T_4) = \alpha\alpha\gamma\alpha$. Similarly, the original table T^O is a leaf in this tree, with sequence $[\gamma, \beta, \alpha, \alpha]$ from T^P . The

q	0	1	2	3	4	5
$\Pr(t(1, 2) = q T^P)$.71	.25	.04	.00	.00	.00
$\Pr(t(1, 4) = q T^P)$.06	.25	.38	.25	.06	.00
$\Pr(t(3, 4) = q T^P)$.00	.71	.25	.04	.00	.00
$\Pr(t(4, 4) = q T^P)$.00	.05	.29	.44	.21	.01

Table 3: Posterior Probabilities for Sensitive Cells

structure of the “forward” tree ($T^O \rightarrow T^P$) and “backward” tree ($T^P \rightarrow T_k$) is shown in Figure 3.

After computing $\Pr(T^P | T_k)$ for each k , the next step is to fix a prior distribution on the T_k . For this example, we take each T_k equally likely. With this assumption (which we examine further below), each T_k has prior probability $1/45$, so we apply Bayes’ rule to get

$$\Pr(T_k | T^P) = \frac{\Pr(T^P | T_k) \Pr(T_k)}{\sum_k \Pr(T^P | T_k) \Pr(T_k)}. \quad (1)$$

Then the probability distribution for each cell of the table is calculated as

$$\Pr(t(i, j) = q) = \sum_{k: t_k(i, j) = q} \Pr(T_k | T^P). \quad (2)$$

For our running example, the four cells originally containing a value of 1 or 2 are of special interest. These are the cells (1,2), (1,4), (3,4) and (4,4). Taking, for example, $\alpha = \beta = 0.25$, their posterior distributions are shown in Table 3. In the table, probabilities in boldface are the published cell values (i.e., those in T^P), while probabilities contained in a box are for the true cell values (i.e., those in T^O).

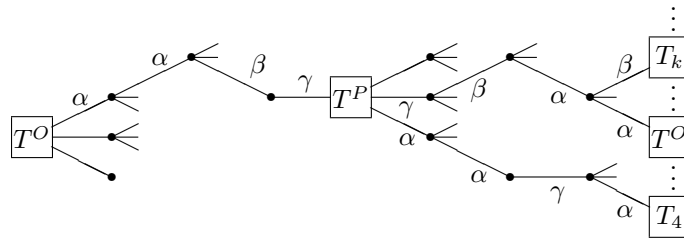


Figure 3: Forward and Backward Tree Traversal

5 Modes of the Posterior Cell Distributions

We note from Table 3 that for each cell in the example above, the probability distribution has its mode at the published value. Thus users need not go through the full analysis of posterior distributions if they are simply interested in determining the most likely value for a cell. This fact remains true under a reasonably general set of assumptions, as we show next.

Theorem 1 *Suppose that M_1, \dots, M_m is a set of cycles such that each cell in T^O is “covered” an equal number of times by the nonzero entries in the cycles, and suppose that $\gamma > \alpha, \beta$. If the prior distribution over the possible tables T_k is uniform, then the mode of the posterior cell distribution $\Pr(t^O(i, j) = q) = \sum_{k: t_k(i, j) = q} \Pr(T_k | T^P)$ coincides with the published cell value $t^P(i, j)$.*

Proof: We sketch the proof for the example given earlier, that is, with the cycles M_1, \dots, M_4 . The full proof follows identical reasoning. Consider first the case where all the cell values $t^P(i, j)$ of T^P are “large,” i.e. at least equal to the number of perturbations possible under the set of cycles. For M_1, \dots, M_4 this is just $t^P(i, j) \geq 2$, since each table cell appears in exactly two of the four data cycles. For equal priors, we have

$$\Pr(T_k | T^P) = \frac{\Pr(T^P | T_k)}{\sum_k \Pr(T^P | T_k)}.$$

The probabilities $\Pr(T^P | T_k)$ for each T_k are determined from the tree in Figure 3. Without loss of generality, consider cell (1, 2). For the cycles M_1, \dots, M_4 cell (1, 2) is unchanged by any of the moves ..CC, ..AB and ..BA (where “..” specifies any of the three outcomes A, B or C in the first two coin flips). Thus the probability of remaining unchanged, $\Pr(t_k(1, 2) = t^P(1, 2))$ is $\gamma\gamma + \alpha\beta + \beta\alpha$. The probabilities $\Pr(t_k(1, 2) = t^P(1, 2) \pm 1, 2)$ are computed in the same way, so for example $\Pr(t_k(1, 2) = t^P(1, 2) + 1) = \gamma\beta + \alpha\gamma$. All of these are less than $\Pr(t_k(1, 2) = t^P(1, 2))$, regardless of the values of α, β and γ . This shows that the mode of the posterior distribution $\Pr(t^O(i, j) = q)$ is the published value.

Now suppose that there are some “small” cells in T^P . When a zero-valued cell prevents a perturbation, one or more subtrees are pruned. Branches marked with γ are never pruned, but depending on the table values, α and β branches may be. If an α branch is pruned, this implies that one or more cells contained in the perturbation must have been negative in the table the α arc leads to. Consequently, the γ branch at the same node must have its probability of traversal increased from γ to 1, since a cell, contained in the data cycle, in the table to the right of the γ arc must be zero. An identical argument shows that if a β arc is pruned, the adjacent γ arc probability increases to 1. The result is a net relative increase in the probability of the cell remaining unchanged from its T^P value. To see this, it is straightforward to check that pruning any combination of non- γ arcs in the subtree to the left in Figure 4 still leaves T^P ’s cell value at the mode of the posterior distribution. For example, when the first α branch is pruned, the tree is that on the right in Figure 4, but the probability of t^P still

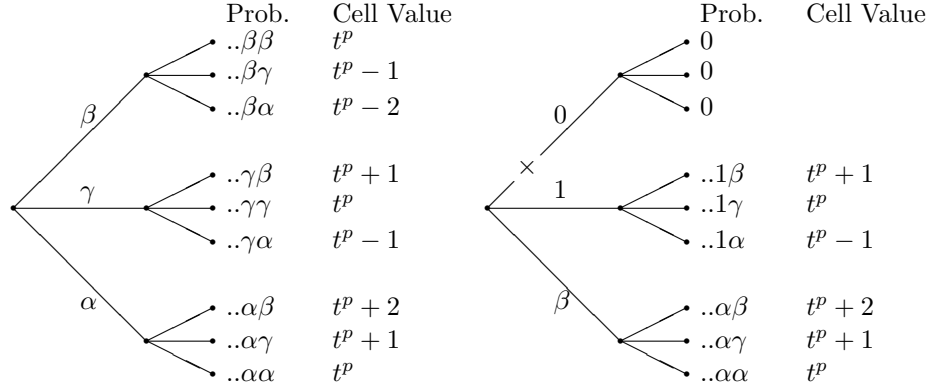


Figure 4: Last two branches, before and after pruning

	Cell value				
	$t^P - 2$	$t^P - 1$	t^P	$t^P + 1$	$t^P + 2$
Prob. before pruning	$..\beta\alpha$	$..\gamma\alpha + \alpha\gamma$	$..\alpha\alpha + ..\gamma\gamma + ..\beta\beta$	$..\gamma\beta + ..\alpha\gamma$	$..\alpha\beta$
Prob. after pruning	0	$..1\alpha$	$..1\gamma + ..\alpha\alpha$	$..1\beta + ..\alpha\gamma$	$..\alpha\beta$

Table 4: Comparison of cell probabilities before and after pruning

dominates the probabilities for other cell values (e.g., $t^P + 1$, etc.), as shown in Table 4. Note that this requires that $\gamma > \alpha, \beta$. This shows that in the case of pruning, the mode of any cell's posterior distribution is still the published value. \square

6 Specifying Prior Distributions

In the example above, we assumed both the data analyst and the data snooper had uniform prior distributions over the set of true tables T_k . But what if one or the other believed that some tables T_k were more likely than others to be T^O ? Abstractly, the analysis only requires that each T_k be assigned a probability, and that these probabilities sum to one. However it seems more reasonable that a person will form her prior distribution over the T_k from knowledge represented in other ways. We examine here a few of the many possibilities.

As an extreme case, suppose the data snooper knows that $T^O(1, 2) = 1$. From the data provider's perspective, this is itself a disclosure, but we can examine the effect of this knowledge on the protection afforded by perturbation to the remaining sensitive cells. It turns out that there are only 18 tables T_k having $T_k(1, 2) = 1$. If we assume that this is the extent of the snooper's prior belief, we might take the prior probability of each of these tables as $1/18$, and the probability of each of the remaining $45 - 18 = 27$ tables to be zero. Repeating

q	0	1	2	3	4	5
$\Pr(t(1, 4) = q T^P)$.07	.30	.35	.26	.02	.00
$\Pr(t(3, 4) = q T^P)$.00	.00	1.00	.00	.00	.00
$\Pr(t(4, 4) = q T^P)$.00	.17	.43	.30	.09	.01

Table 5: Posterior Probabilities for Sensitive Cells, Knowing $T^O(1, 2) = 1$

	$PV - 2$	$PV - 1$	PV	$PV + 1$	$PV + 2$
Shape A	.05	.29	.44	.21	.01
Shape B	.00	.00	.71	.25	.04
Shape C	.04	.29	.46	.21	.00
Shape D	.06	.25	.38	.25	.06

Table 6: Posterior Probabilities Shapes, Where $PV =$ Published Value

Equations 1 and 2 for each cell, we arrive at the posterior distributions for the remaining sensitive cells shown in Table 5.

Table 5 immediately points out a consequence of the choice of data cycles. If the snooper knows for certain that $T^O(1, 2) = 1$, this knowledge compromises the value of $T^O(3, 4)$. The reason for this can be traced to the set of perturbations M_1, \dots, M_4 used to disguise the data. Examining these again, it's clear that the perturbations for this pair of cells move in lockstep; knowing one reveals for certain the value of the other.

In fact, with this set of perturbations, and the case of uniform priors, there are only four different posterior cell distribution shapes. To find the actual posterior distribution for a cell, it is enough to know its shape and the cell value in the published table. The distribution shape is then translated so that its mode is centered over the published value. The shapes are given in Table 6 and their locations in the table are as follows.

$$T^P = \begin{bmatrix} A & B & C & D \\ D & A & B & C \\ C & D & A & B \\ B & C & D & A \end{bmatrix},$$

This result in no way diminishes the usefulness of cyclic perturbation, since each cell receives adequate protection. It simply says that several cells will have the same shape for its distribution. There is an upper limit on the number of different cell distributions possible. This is a consequence of the fact that are only so many degrees of freedom available when table margins are fixed.

Another example of specifying priors is similar. A data snooper might have been part of the survey, and would of course know his own attribute values, placing him in, say, cell (1,2). So the snooper would have certain knowledge that the count in cell (1,2) was at least one. Once again, if the snooper knew

q	0	1	2	3	4	5
$\Pr(t(1, 2) = q T^P)$.00	.86	.14	.00	.00	.00
$\Pr(t(1, 4) = q T^P)$.06	.25	.38	.25	.06	.00
$\Pr(t(3, 4) = q T^P)$.00	.00	.86	.14	.00	.00
$\Pr(t(4, 4) = q T^P)$.00	.18	.43	.32	.07	.01

Table 7: Posterior Probabilities for Sensitive Cells, Knowing $T^O(1, 2) \geq 1$

nothing more, he might place equal prior probability on the 27 tables T_k having $T_k(1, 2) \geq 1$. The cycle-induced coupling between cells illustrated above persists in this case, but to a lesser degree. The posterior distributions for the sensitive cells are given in Table 7.

7 Measuring Data Utility and Disclosure Risk

Statistical disclosure limitation has the twin goals of maintaining adequate data utility to legitimate users while at the same time reducing disclosure risk to an acceptable level. In this section, we propose several measures for both data utility and disclosure risk, and examine cyclic perturbation under the lens of these measures. In Section 8, we compare cyclic perturbation with two well-known disclosure limitation methods, cell suppression and controlled rounding.

7.1 Data Utility

Data utility is defined as a measure of the value of information to a legitimate user or class of users. A user wishes to make inferences from the disclosure-limited data that are as close as possible to those she would make if she were in possession of the actual data. Ideally, it should be possible for a user to determine the magnitude of errors that could be made in analyzing the released data rather than the actual data. The information organization is in a position to assess the true data utility of a release, since it knows the actual data collected. However, the information organization cannot know all the possible uses to which the data may be put, nor can it know a user's priors over the possibilities for the true data. Thus a measure of data utility relies critically on assumptions made about the intended use, and users, of the data.

Users may be interested in the likely values of individual table cells, or may be interested in statistics that are derived from the table as a whole. For the former, a comparison of the true cell value with the user's posterior probability distribution is reasonable. One such measure is mean squared precision, the reciprocal of mean squared error based on the distribution of cell values and the knowledge of the true cell value. For the table as a whole, a comparison of summary statistics of association between row and column variables (the χ^2 statistic for a test of independence, or perhaps better, its p -value) would give

0.71	1.33	0.50	0.50
0.50	0.71	1.33	0.50
0.50	0.50	0.71	1.33
1.33	0.50	0.5	0.71

Table 8: Mean square precision for cells in the perturbed table

another indication of how the disclosure limited table differs from the true table.

Any of these measures of data utility can be evaluated for cyclic perturbation, either for a specific perturbed table or, with more effort, for the procedure in general. The fact that $\Pr(T^P | T_k)$ for all k can be computed in a straightforward way confers a distinct advantage over other disclosure limitation methods.

As a concrete demonstration, mean squared precision (MSP) is calculated for our running example. Table 8 shows the MSP values for cells in the 4×4 table. Later we compare these values with the familiar disclosure limitation methods of cell suppression and controlled rounding.

7.2 Disclosure Risk

Disclosure risk measures the degree to which the confidentiality promised to a data subject might be compromised by a data release. Since confidentiality is most often concerned with individuals, measures of disclosure risk for tables commonly focus on the accuracy with which particular cell counts can be determined. In contrast to cell-specific measures of data utility, however, disclosure risk is not necessarily solely concerned with the accuracy with which a data intruder can *correctly* identify sensitive information about a respondent. While such concerns are obviously important, many data providers also recognize that it is important to prevent a data snooper from assigning a high probability to any cell value, even if the high probability value is incorrect (Duncan and Lambert 1986).

A general framework for disclosure risk is given in (Duncan, Fienberg, Krishnan, Padman and Roehrig 2001). It models the risk for any individual table cell as

$$\sum_k r(k)p(k)$$

where $r(k)$ is the risk associated with a data snooper obtaining knowledge that a cell entry has true value k , and $p(k)$ is the probability that the cell value is k , given the released table T and the knowledge held by the data snooper about the disclosure-limiting method employed.

Two realizations of this framework are the following. First, since cell values of 1 or 2 are traditionally considered revealing, we might consider $\Pr(x_{ij} = 1 \text{ or } 2 | T)$. This corresponds to

$$r(k) = \begin{cases} 1 & \text{if } x_{ij} = 1 \text{ or } 2 \\ 0 & \text{otherwise} \end{cases}$$

0.80	1.38	0.85	0.71
0.71	0.80	1.38	0.85
0.85	0.71	0.80	1.38
1.38	0.85	0.71	0.80

Table 9: Risk (entropy measure) for cells in the perturbed table

Another approach is to use an entropy measure such as

$$-\sum_k \log(p(k))p(k).$$

Here, $r(k) = \ln(p(k))$ and the final disclosure risk measure is normalized to

$$\text{Risk} = \frac{1}{-\sum_k \ln(p(k))p(k)}.$$

Both of these measures can be easily be computed within the framework of cyclic perturbation. For example, the values of the entropy measure for our running example are given in Table 9. Table 9 shows that some cells are more vulnerable to data snooper attack than others. Most vulnerable are the four cells (1,2), (4,1), (2,3) and (3,4).

8 Comparison With Other Disclosure Limitation Methods

8.1 Publishing Only the Marginals

If a table contains many small cell values, it might be considered too great a risk to publish even a disclosure-limited version of it. Thus is is not uncommon to “punt” and decide to reveal only the table’s marginal totals—an extreme version of cell suppression. In many situations in which a particular table is part of a larger hierarchical structure, this can seem reasonable, since the completely suppressed table entries are correctly summarized by their inclusion into a table at a higher level in the hierarchy.

Applying the methodology of De Loera and Sturmfels (2001), we found that the number of tables that have the marginals in our example is over 18 billion, specifically 18,272,363,056. On the face of it, this would suggest that the table is quite well protected. On the other hand, from a Bayesian perspective, not all of these tables would typically be equally likely. Perhaps more to the point, a data intruder is likely interested in the possible values for individual cells, and so the total number of distinct tables may not be wholly relevant. We next present two types of analysis for the margins-only problem.

If we consider the given marginal totals to be constraints in an integer programming problem, then by minimizing and maximizing each cell in turn, the

tightest cell bounds can be found. Narrow bounds (especially the limiting case where upper and lower bounds are equal) indicate a disclosure. For the case of two-dimensional tables, one only needs to use the linear programming relaxation of the integer program. Indeed, several network procedures are available (Gusfield 1988, Gusfield 1990). For three- and higher-dimensional tables, the situation is much worse. De Loera and Ohn (2001) show that the bounds problem for $4 \times 4 \times 4$ and larger tables is *NP*-complete.

It is also possible to estimate the cell distributions in the case where only the margins are given. Following the MCMC procedure of Diaconis and Sturmfels (1998), one can sample from the space of all tables agreeing with the margins (for our example, the set of 18 billion tables mentioned above). With appropriate priors on these tables, a Bayesian analysis along the lines of the one for cyclic perturbation can be done, but it will take several days of computation for even a 4×4 table with current (2004) desktop computing power.

For larger tables, and especially for higher-dimensional tables, the computational complexity of these procedures is daunting. For example, the set of Markov transitions required for the MCMC estimation procedure of Diaconis and Sturmfels must first be computed. Formally, this is known as a Gröbner basis, and for finding this basis a well-defined algorithm exists (Buchberger's algorithm). However, one fast implementation of this algorithm runs in about 25 milliseconds for a $3 \times 3 \times 3$ table, 20 minutes for the $4 \times 3 \times 3$ case, and *three months* for the $5 \times 3 \times 3$ case, on a fast 2004 PC. These times are for just one implementation, and better algorithms may give improvements, but the overall trend is clear.

8.2 Cell Suppression

A table cell determined to be vulnerable must somehow be disguised, and an obvious way of accomplishing this is to completely suppress its value. This is shown below for our running example by replacing the true cell value with s .

	w_1	w_2	w_3	w_4	
v_1	15	s	3	s	20
v_2	20	10	10	15	55
v_3	3	10	10	s	25
v_4	12	14	7	s	35
	50	35	30	20	135

These are called *primary* suppressions. For most patterns of primary suppressions, and our example is no exception, it is possible to recover some or all suppressed cell values by simple subtraction, using the marginal totals. To avoid this possibility, additional non-sensitive cells are also suppressed. These are called *complementary* suppressions, and are usually found using an algorithm designed to prevent recovery of a primary suppression while simultaneously minimizing the total number of suppressions.

	w_1	w_2	w_3	w_4	
v_1	15	s	s	s	20
v_2	20	10	10	15	55
v_3	3	10	s	s	25
v_4	12	s	7	s	35
	50	35	30	20	135

Figure 5: Table With Cell Suppressions

Different agencies may use different rules to determine which cells are sensitive, and also which cells are candidates for complementary suppression. As an example of the latter, some agencies do not permit cells with true values of zero to be suppressed, since it may be common (or easily available) knowledge that such a cell must be a zero (i.e., a structural zero). Additionally, the level of required protection, at the individual cell level, may vary. For example, some IOs require only that the exact value of a sensitive cell be ambiguous. Others require that the suppression patterns guarantee that a wider range of sensitive cell values be possible. For any cell, the range of possible values under a suppression pattern is called its *feasibility interval*. Any discussion of disclosure risk and data utility for tables subject to cell suppression will necessarily depend on the details of the specific protection requirements imposed.

Assuming as before that cells with values of one or two are sensitive, one possible pattern of primary and complementary suppressions that does protect confidentiality is shown in Figure 5.

8.2.1 Disclosure Risk and Data Utility for Cell Suppression

To analyze the disclosure risk associated with the release of a table T^S with suppressions, we might try to repeat the steps used in the analysis of cyclic perturbation. This first involves enumerating those tables T_k that could have resulted in T^S , had they been subjected to cell suppression. This in turn depends on the rule used for identifying primary suppressions, and any rules restricting complementary suppressions and requiring specific feasibility intervals. Supposing that the disseminator chooses exactly those cells containing a one or two as primary suppressions, we can draw a network to facilitate the enumeration process, taking the rule into account. For the complementary suppression pattern shown above, the network is that of Figure 6.

In the network shown in Figure 6, nodes on the left represent rows, while those on the right represent columns. Directed arcs in the central portion of the network represent the suppressed cells, and any specific set of flows along these arcs represents a possible set of suppressed cell values. The arrows entering the v nodes are sources which must supply the quantities given, while the arrows leaving the w nodes are sinks which draw the quantities shown. The source

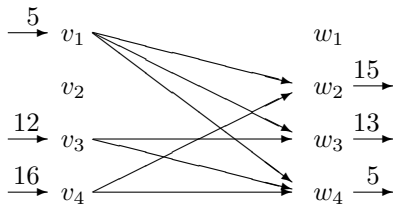


Figure 6: Network for Cell Suppression

and sink values are determined from the marginal totals, after subtracting away every unsuppressed value. Since the tables we are considering have non-negative integer values only, any feasible table corresponds to a total flow in which all the central arcs carry non-negative integer values.

Consider the case where (1) primary suppressions are those cells containing a one or two, and (2) complementary suppressions are chosen only from the remaining nonzero cells. Analysis of the network shows that there are only three tables T_k that, under these conditions, could have been the true table T^O . These are shown without marginal totals:

15	1	2	2
20	10	10	15
3	10	11	1
12	14	7	2

15	1	3	1
20	10	10	15
3	10	10	2
12	14	7	2

15	2	2	1
20	10	10	15
3	10	11	1
12	13	7	3

The reason that there are so few potential tables can be traced to the fact that the sum of the suppressed cells in the first row is 5. Given the rule that cells containing zeros are not suppressed, there are very few choices for these cells. Further limiting the possibilities is the fact that some network flows (corresponding to possible tables) have all the values in some cycle greater than 2. Since, by our assumption, primary suppressions consist only of cells containing a 1 or a 2, such a table could not have been the original, unsuppressed table.

Carrying out the risk analysis presented earlier requires first that the conditional probabilities $\Pr(T^S | T_k)$ be determined for each k . Unlike the situation for cyclic perturbation, there seems to be no clearcut way of deciding on these values. The exact and heuristic cell suppression procedures we have examined (Cox 1995, Kelly, Golden and Assad 1992, Fischetti and Salazar 1998) are deterministic, but highly complex, often using the simplex method of linear programming as the underlying technique. If one were in possession of the exact implementation of the cell suppression algorithm used by the data disseminator, it would be possible to assign a value, either 0 or 1, to $\Pr(T^S | T_k)$.

For example, if running the IO's cell suppression algorithm on the three tables $T_k, k = 1, 2, 3$ above determined that each of them resulted in the published

∞	3	3	3
∞	∞	∞	∞
∞	∞	3	3
∞	3	∞	3

Table 10: Mean square precision for cells in the suppressed table

∞	3.29	3.29	3.29
∞	∞	∞	∞
∞	∞	3.29	3.29
∞	3.29	∞	3.29

Table 11: Entropy-based disclosure risk for cells in the suppressed table

table T^S , then for any prior probabilities $\Pr(T_k)$ we would have $\Pr(T_k | T^S) = \Pr(T_k)$. But if it turned out that applying the suppression routine produced T^S for $k = 1, 2$ but a different table for $k = 3$, we would conclude that $\Pr(T_1 | T^S) = \Pr(T_1)/(\Pr(T_1) + \Pr(T_2))$ and $\Pr(T_2 | T^S) = \Pr(T_2)/(\Pr(T_1) + \Pr(T_2))$. In this case, it is certain that, for example, $t^O(1, 2) = 1$, and this is a definite disclosure.

If we assume that each of the three tables $T_k, k = 1, 2, 3$ above did in fact result in the suppression pattern above, then the corresponding tables for data utility and disclosure risk appear in Tables 10 and 11 respectively.

8.3 Controlled Rounding

Controlled rounding (Cox 1987, Kelly, Golden, Assad and Baker 1990) produces from T^O a published table T^R whose entries and marginal totals are all close to, but not necessarily equal to, the true cell and marginal values of T^O . Specifically, the IO chooses a *rounding base* b , typically in the range of 3 to 5, and then determines a rounded table in which each cell and marginal total equals an adjacent integer multiple of b . Controlled rounding is a rounding in which cell values that are initially either zero or already an integer multiple of b are left unchanged. Controlled rounding for two-way tables is quite easy. For three- and higher-dimensional tables, a controlled rounding may not exist (Kelly, Golden, Assad and Baker 1990).

One controlled rounding, to base 3, of our running example is the following.

	w_1	w_2	w_3	w_4	
v_1	15	0	3	0	18
v_2	21	9	12	15	57
v_3	3	9	9	3	24
v_4	12	15	6	3	36
	51	33	30	21	135

The number of tables T_k that could have produced this rounded table T^R is surprisingly large; a brute-force enumeration shows it to be more than one billion. Nonetheless, one might hope that an analysis of expected utility and disclosure risk could be performed. However, it is not completely clear how to assign, as we have done for cyclic perturbation, the conditional probabilities $\Pr(T_R | T^k)$. These probabilities depend on the explicit sequence of steps in the rounding algorithm. Many controlled rounding procedures exist, and to our knowledge, none of the algorithms presented in the literature is described in sufficient detail to allow such an analysis. For example, Cox's 1987 procedure includes instructions such as "choose any fraction c_{ij} in C " (where C is a special table defined for his process). Inevitably, implementations of this algorithm will differ due to the data structures chosen, and even programmer preference. Each correct implementation will produce a correctly rounded table, yet two such implementations may produce *different* rounded tables.

To investigate controlled rounding further, we wrote an implementation of the Cox procedure. This particular implementation does not include the stochastic step required to achieve unbiasedness (Step 4, p. 522), so the algorithm is well-defined and repeatable. We looked for tables that did round, via our implementation, to the rounded table given above. There were 17,132,236 such tables, suggesting the obvious point that once a specific implementation is chosen, the number of tables that round into a given published table is much smaller than the number that could, with *some* implementation round into it. Since our implementation had no stochastic steps, $\Pr(T_R | T^k) = 1$ for each of these tables.

As in the analysis of cyclic perturbation, we are free to choose any prior distribution over the tables or cell entries. For example, if we are willing to assume that the prior probabilities over the T_k are equal, then something may be said about T^O cell probabilities, by simply examining the T_k and tallying the proportions of each cell value. For this choice of prior, we found, for example, that cell (1,2) had the posterior probability 0.482 for the value 0, probability 0.338 for the value 1, and probability 0.180 for the value 2. For every cell, every posterior probability was less than 0.5, and typically the largest posterior probability was no greater than 0.33.

Even for our 4×4 table, the analysis of controlled rounding is difficult, since it requires the construction of many millions of tables. The results presented here took several days of computer time to produce. The analysis of the full unbiased controlled rounding procedure recommended by Cox would be even more difficult, since a tree structure essentially identical to that used above for

0.90	1.51	0.72	1.36
0.42	0.42	0.16	0.72
0.76	0.38	0.48	0.44
0.73	0.51	0.46	0.40

Table 12: Mean square precision for cells in the rounded table

0.96	0.97	0.70	1.17
0.67	0.66	0.66	0.69
0.66	0.66	0.65	0.66
0.66	0.66	0.65	0.65

Table 13: Entropy-based disclosure risk for cells in the rounded table

cyclic perturbation, would be needed. From a data user’s perspective, the sheer number of possible tables T^O makes a complete Bayesian analysis of the data impractical.

Table 12 is the table of mean squared precision values for our running example, using controlled rounding, and Table 13 shows cell risk using the entropy measure.

Figure 7 shows the results of our analysis for cyclic perturbation, cell suppression and controlled rounding, for the cell (3,3) in our example, while Figure 8 compares values for cell (3,3).

9 Conclusions

Cyclic perturbation is an intuitively attractive perturbation method for limiting disclosure risk with tabular data. It permits a tractable Bayesian analysis that can be used to develop an R-U confidentiality map. This information permits an information organization to examine tradeoffs between disclosure risk and data utility.

In addition to providing details of cyclic perturbation, we have compared it, from a risk-utility perspective, with the conventional techniques of cell suppression and controlled rounding. It is important to note that the computation required for generating the R-U confidentiality maps for the latter two methods is two or three orders of magnitude greater than that required for cyclic perturbation. Thus in practice, with larger tables, an IO will simply not know the risk-utility tradeoff resulting from the use of either of the conventional techniques.

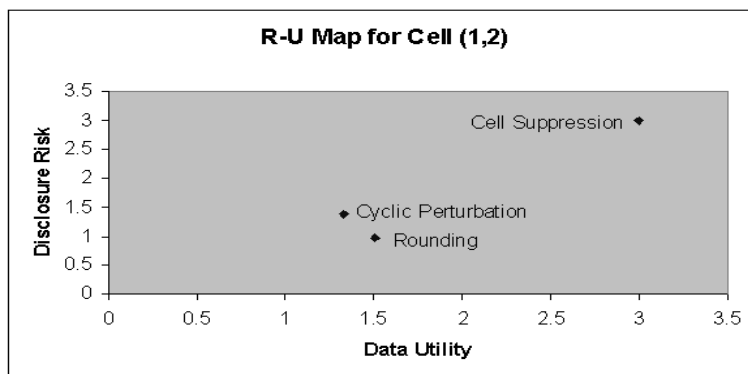


Figure 7: The R-U confidentiality map for our problem

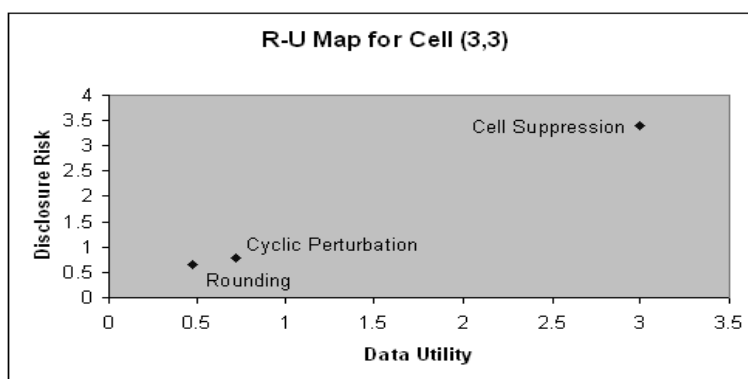


Figure 8: The R-U confidentiality map for our problem

References

- [1] Cox, L.H. (1980). Suppression Methodology and Statistical Disclosure Control. *Journal of the American Statistical Association*, 75, pp. 377–385.
- [2] Cox, L.H. (1995). Network Models for Complementary Cell Suppression. *Journal of the American Statistical Association*, 90, pp. 1453–62.
- [3] Cox, L.H. (1987). A Constructive Procedure for Unbiased Controlled Rounding. *Journal of the American Statistical Association*, 82, pp. 520–524.
- [4] Cox, L.H., McDonald, S. and Nelson, D. (1986). Confidentiality Issues of the U.S. Bureau of the Census. *Journal of Official Statistics*, 2, pp. 135–160.
- [5] De Loera, J. and Onn, S. (2001). The Complexity of Three-Way Statistical Tables. Preprint, Department of Mathematics, University of California, Davis.
- [6] De Loera, J. and Sturmfels, B. (2001). Algebraic Unimodular Counting. Preprint, Department of Mathematics, University of California, Davis.
- [7] Diaconis, P. and Sturmfels, B. (1988). Algebraic Algorithms for Sampling from Conditional Distributions. *Annals of Statistics*, 26, pp. 363–397.
- [8] Duncan, G., Fienberg, S., Krishnan, R., Padman, R. and Roehrig, S. (2001). Disclosure Limitation Methods and Information Loss for Tabular Data. In *Confidentiality, Disclosure, and Data Access: Theory and Practical Applications for Statistical Agencies*, P. Doyle, J. Lane, J. Theeuwes and L. Zayatz, eds., Amsterdam: North-Holland.
- [9] Duncan, G. and Fienberg, S. (1998). Obtaining Information While Preserving Privacy: A Markov Perturbation Approach. In *Proceedings of Statistical Data Protection '98*, March 1998, Lisbon.
- [10] Duncan, G. and Lambert, D. (1986). Disclosure-Limited Data Dissemination (with Discussion). *Journal of the American Statistical Association*, 81, 10–28.
- [11] Duncan, G., Jabine, T. and de Wolf, V. (1993). *Private Lives and Public Policies: Confidentiality and Accessibility of Government Statistics*. Washington, D.C: National Academy Press.
- [12] Duncan, G. and Pearson, R. (1991). Enhancing Access to Data While Protecting Confidentiality: Prospects for the Future. Invited paper with discussion by L. Cox, S. Keller-McNulty and J. Norwood. *Statistical Science* 6, pp. 219–239.
- [13] Fienberg, S., Makov, U. and Steel, R. (1998). Disclosure Limitation Using Perturbation and Related Methods for Categorical Data. *Journal of Official Statistics*, 14, 485–502.

- [14] Fischetti, M. and Salazar, J.J (1998). Modeling and Solving the Cell Suppression Problem for Linearly-Constrained Tabular Data. In Proceedings of Statistical Data Protection '98, March 1998, Lisbon.
- [15] Gopal, R.D., Goes, P.B. and Garfinkel, R. (1998) Interval Protection of Confidential Information in a Database. *INFORMS Journal On Computing*, 10, pp. 309–322.
- [16] Gusfield, D. (1990). A Little Knowledge Goes a Long Way: Faster Detection of Compromised Data in 2-D Tables. Proceedings of the 1990 IEEE Conference on Research in Security and Privacy, pp. 86-94.
- [17] Gusfield, D. (1988). A Graph Theoretic Approach to Statistical Data Security. *SIAM Journal on Computing*, 17, p. 552–571.
- [18] Hawala, S., Zayatz, L. and Rowland, S. (2004). American FactFinder: Disclosure Limitation for the Advanced Query System. *Journal of Official Statistics* 20, pp. 115–124.
- [19] Kelly, J.P., Golden, B.L. and Assad, A.A. (1992). Cell Suppression: Disclosure Protection for Sensitive Tabular Data. *Networks*, 22, pp. 397-417.
- [20] Kelly, J.P., Golden, B.L., Assad, A.A. and Baker, E.K. (1990). Controlled Rounding of Tabular Data. *Operations Research*, 38, pp. 760-772.
- [21] de Wolf, P-P., Gouweleeuw, J., Kooiman, P. and Willenborg, L. (1998) Reflections on PRAM. In Proceedings of Statistical Data Protection '98, March 1998, Lisbon.
- [22] Subcommittee on Disclosure Limitation Methodology (1994). Report on Statistical Disclosure Limitation Methodology. Working Paper #22 Federal Committee on Statistical Methodology, Office of Management and Budget.
- [23] Willenborg, L. and De Waal, A. (1996). *Statistical Disclosure Control in Practice*. New York: Springer-Verlag.