MEMORYLESS INFERENCE CONTROLS FOR STATISTICAL DATABASES

Dorothy E. Denning[1]
SRI International

Jan Schlörer and Elisabeth Wehrle[2]
Universität Ulm

August 1982
Revised: March 1984

Abstract. Memoryless inference controls that protect confidential data released as summary statistics about subgroups of individuals are studied. For each new query, the database attempts to determine whether release of the statistic could lead to compromise; this is done without keeping a record of previous queries or a list of permitted statistics. Using a model that structures the set of all possible statistics into a lattice of statistical tables, we show that a memoryless control that suppresses at the cell level in tables, thereby permitting partial tables, is practical only if the syntax of queries is restricted. With an unrestricted syntax, it is more practical to suppress complete tables. Several memoryless table suppression techniques are described and evaluated.

[1] Author's present address: SRI International, 333 Ravenswood Ave., Menlo Park, CA 94025; research supported in part by NSF grant MCS 80-15484 while the author was at Purdue University, and NSF grant MCS 83-13650.

[2] Authors' address: Universität Ulm, Klinische Dokumentation, Eythstr. 2, D-7900 Ulm, W. Germany; research supported by Stiftung Volkswagenwerk.

1. Introduction

Statistical databases aim to provide frequencies, averages, and other statistics about groups of persons (or organizations), while protecting the confidentiality of the individuals represented in the database. This objective is difficult to achieve, as users of statistical databases have a host of inference techniques at their disposal for retrieving information about identifiable persons (e.g., see [DeJo83 , Denn79a, Denn80a, Denn82a, Dobk79a, Reis78a, Schl80a].)

There are two broad categories of inference controls: controls that place restrictions on the set of allowable queries, suppressing those that are not allowed; and controls that add noise to the data or to the released statistics. This paper focuses on restriction (suppression) techniques, though we shall briefly discuss how these techniques can be strengthened with perturbation techniques.

A statistical database is modeled as a lattice of statistical tables, where the statistics computed over groups of records having m attribute values in common correspond to cells (or cell unions) of m-dimensional tables in the lattice. Output restriction techniques are classified according to whether they restrict at the table level or cell level in the lattice, and according to whether they are a priori, audit based, or memoryless. Whereas **table-level** controls suppress complete m-tables of statistics, **cell-level** controls aim to suppress as few cells in a table as possible.

A priori controls determine in advance a fixed set of statistics that can be released without compromising any individual's privacy. These statistics are typically released as 1- or 2-dimensional tables of counts or sums. The cell suppression techniques used by census agencies [Cox78a, Cox80a, Sand77a] are a priori cell-level controls. Partitioning [Chin79a, Chin81a, Yu77a, Schl83] is an a priori control with both cell-level and table-level characteristics.

Audit-based controls keep a history of queries to determine whether release of a statistic, when correlated with previously released statistics, could lead to compromise. A query-set-overlap control [Dobk79a], Fellegi's auditing control [Fell72a], and Chin's and Ozsoyoglu's audit expert [Chin82a] are audit-based cell-level controls.

Memoryless controls attempt to determine whether releasing a statistic leads to compromise without keeping a record of previous queries or a list of permitted statistics. A query-set-size control [Hoff70a] is an example of an easily subverted memoryless cell-level control. Although both audit-based and memoryless controls determine whether to release a statistic at query processing time, memoryless controls are potentially more efficient, requiring neither space nor time to process audit records. The objective of this paper is to examine the feasibility of providing a memoryless control powerful enough to control disclosure, while not being unnecessarily restrictive.

Section 2 reviews the lattice model, which is the basic structure used by census agencies for analyzing potential disclosures. Because the model is fundamental to understanding the concepts of cell-level and table-level controls, it is presented as a tutorial. Section 3 defines disclosure of sensitive statistics.

Section 4 draws on earlier work to examine the feasibility of a secure memoryless cell-level control. For a given query over m attributes, we investigate how many cells in the m-dimensional table defined by the values of these attributes must be examined in order to determine whether release of the statistic is secure. We show that a memoryless cell-level control is practical only if the syntax of queries is restricted, and introduce a new heuristic approach to cell suppression based on such a syntax.

Section 5 investigates simple criteria for suppressing complete tables of statistics without examining the cells of a table. We evaluate and compare four criteria for deciding which tables to suppress. Two of the criteria are practical and effective: one is based on the relative size of a table; the other on explicit risk estimation. Most of the new results of this paper are contained in this section.

Because the memoryless table suppression criteria do not guarantee security, they must be supplemented with other simple controls. Section 6 briefly outlines techniques for strengthening table-level controls. Section 7 discusses the problem of tuning the restriction criterion for different types of statistics.

We shall assume a database is static, and not discuss the problems caused by insertions, deletions, and modifications (see [Chin79a, Chin81a, Ozso81a, Yu77a]).

2. Lattice Model

For statistical purposes, a database can be viewed as a collection of N **logical records**, each describing an individual or organization. The i^{th} record $(1 \le i \le N)$ contains values $x_{i1},...,x_{iM}$ for M attributes $A_1,...,A_M$. Each **attribute** (or **variable**) A_k has $|A_k|$ possible values in its domain. Some attributes have nonnumeric values; an example is SEX, whose two possible values are MALE and FEMALE. Others have numeric values; an example is a student's grade-point.

The model describes neither the database schema nor its implementation, but rather a conceptual view of the data in the database (e.g., see [Olss75a, Rapa75a, Sund73a, Sund78a]). Chin and Ozsoyoglu [Chin81a] have modeled a related structure or view of the data within the framework of some existing conceptual data models.

Statistics are computed for subsets of records having common attribute values. A set of records is specified by a **characteristic formula** C, which, informally, is any logical formula over the values of the attributes using (in increasing priority) the operators OR (+), AND (&), and NOT (~). An example of a formula is

$$C = (\mathrm{SEX}=\mathrm{MALE})\ \&\ [(\mathrm{MAJOR}=\mathrm{CS}) + (\mathrm{MAJOR}=\mathrm{EE})]\ \&\ (\mathrm{CLASS}=1983)\ , \quad (2.1)$$

which, for a student database, specifies all male students majoring in either CS or EE and belonging to the class of 1983.

The set of records satisfying a characteristic formula C is called the **query set** of C. We shall write "C" to denote both a formula and its query set, and $|C|$ to denote the number of records in C (i.e., the size of C). We denote by "ALL" a formula whose query set is the entire database; thus $|\mathrm{ALL}| = N$, and $C \subseteq \mathrm{ALL}$ for any formula C, where "$\subseteq$" denotes query set inclusion. A query set for a formula over m distinct attributes is called an **m-set** (assuming the formula cannot be simplified to one having fewer than m attributes). The query set C in (2.1), for example, is a 3-set. Note that our concept of a query set differs from the usual set concept. A query set is an object determined by its characteristic formula and not just by the records comprising the set. If the normal forms (e.g., minterms) of two formulas differ, their query sets are distinct objects even if both contain the same records. This remains true even if no records satisfy a formula. The conventional set definition may entail serious security problems [Denn83, Schl84].

An **elementary m-set** is an m-set specified by a conjunctive formula of the form

$$E = (A_1 = a_{i_1})\ \&\ \cdots\ \&\ (A_m = a_{i_m})\ , \quad (2.2)$$

where $A_1,\ldots,A_m$ are attributes and each a_{i_j} is some value in the domain of A_j. A query set of this form is called an elementary set because it cannot be decomposed without introducing additional attributes. Note that all m-sets over $A_1, \ldots, A_m$ can be expressed as unions of the elementary sets of the attributes. The query set C in (2.1), for example, is the union of the elementary

3-sets

(SEX=MALE) & (MAJOR=CS) & (CLASS=1983)

(SEX=MALE) & (MAJOR=EE) & (CLASS=1983) .

Given $A_1, \ldots, A_m$, the total number of elementary m-sets is

$$s_m = \prod_{j=1}^{m} |A_j| \ .$$

These s_m sets define an m-dimensional table or **m-table** T, which partitions the database into s_m query sets, where each attribute A_j corresponds to one dimension of the table. Figure 1 illustrates a 2-table over SEX and MAJOR; each entry in the table gives the number of records falling into its respective elementary set.

		MAJOR			
		ART	BOT	CS	EE
SEX	MALE	10	9	11	12
	FEMALE	5	1	9	8

Figure 1. 2-Table by SEX and MAJOR.

Note that a table need not correspond to a physical structure of the database. A database with M attributes has 2^M such tables, corresponding to all possible subsets of the attributes. There is exactly one M-table, where the records in each elementary M-set are indistinguishable (except possibly for a statistically irrelevant identifier field).

The set of 2^M tables $T_1, \ldots, T_{2^M}$ form a **lattice** with partial ordering relation "$\leq$", where $T_i \leq T_j$ means each elementary set in table T_j corresponds to a union of elementary sets in table T_i; thus, T_i is a refinement of T_j. Figure 2 shows the lattice of tables defined over $M = 4$ attributes A, B, C, and D. We have, for instance, ABCD $\leq$ ABD $\leq$ AB $\leq$ B $\leq$ ALL. An elementary 2-set (A=a) & (B=b) in

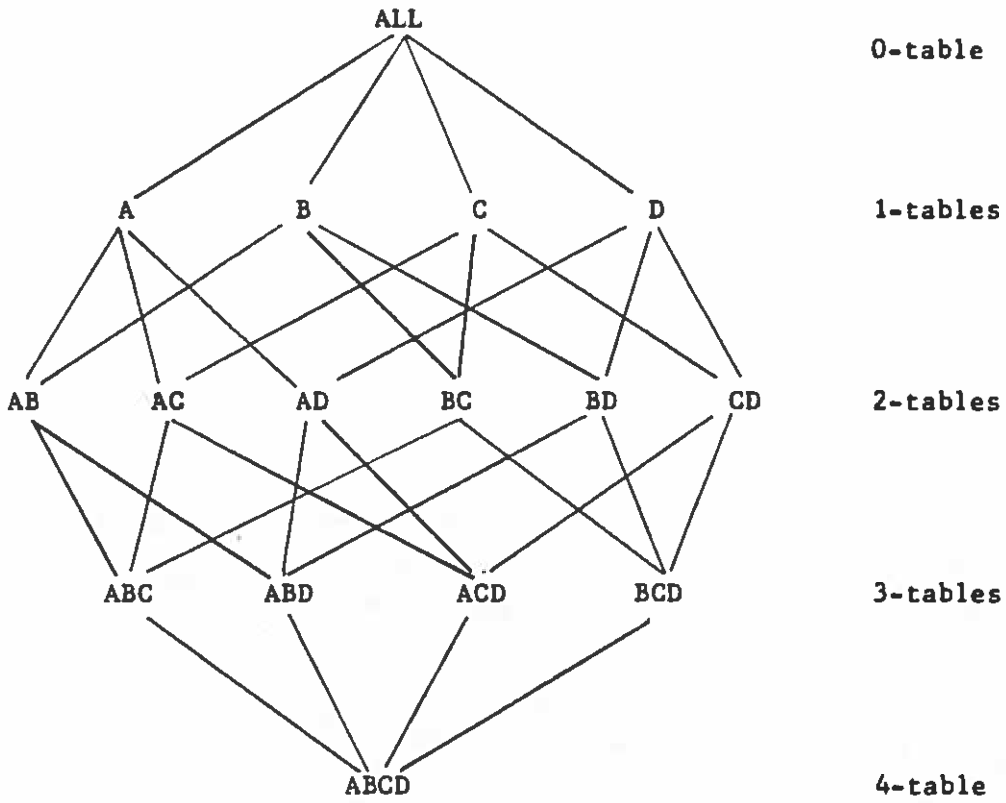

Figure 2. Lattice of Tables Over Attributes A, B, C, and D.

table AB, for example, is a union of elementary 3-sets over attribute C in table ABC or over attribute D in table ABD. Using table ABD, we have

$$(A=a) \ \& \ (B=b) = \bigcup_{d} (A=a) \ \& \ (B=b) \ \& \ (D=d) \ ,$$

where the union is taken over all values d in the domain of D. The elementary 3-sets of table ABD in turn are unions of elementary sets in table ABCD. The single elementary 0-set ALL is the union of all elementary sets in any other table.

In a student database, for example, the elementary 1-set (SEX=FEMALE) in the 1-table defined by SEX is the union of elementary sets over (SEX=FEMALE) and some other attribute, such as MAJOR or CLASS. For MAJOR, we have

$$(\text{SEX}=\text{FEMALE}) = \bigcup_{deptname} (\text{SEX}=\text{FEMALE}) \ \& \ (\text{MAJOR}=deptname) \ .$$

Statistics are calculated over values in the records belonging to a query set C, and have the form $f\,(C,D)$, where D is a (possibly empty) set of attributes and f is a statistical function. An attribute is called a **characteristic attribute** if it appears in C and a **data attribute** if it appears in D; an attribute can appear in both.

By $q\,(C)$ we shall mean an **additive statistic** $f\,(C,D)$, or **query** for an additive statistic, with the following property: If $C = C_1 \cup \cdots \cup C_k$ for pairwise disjoint query sets $C_1, \ldots, C_k$, then

$$q\,(C) = q\,(C_1) + \cdots + q\,(C_k) \ . \qquad (2.3)$$

Because any m-set C over attributes $A_1, \ldots, A_m$ is a union of elementary m-sets over $A_1, \ldots, A_m$, property (2.3) implies that $q\,(C)$ can be computed by adding the statistics for the elementary m-sets comprising C, assuming these statistics are released.

Many statistics are additive, or easily computed from additive statistics [Dale82a]. **Counts** (absolute frequencies or cardinalities) and **sums** are additive, defined by

$$\text{COUNT}(C) = |C| , \qquad \text{SUM}(C, A_j) = \sum_{i \in C} x_{ij} ,$$

where A_j is an attribute having numerical values, and x_{ij} is the value of A_j in record i. For example, the statistic COUNT((SEX=MALE) & (MAJOR=CS)) gives the number of males majoring in CS; the statistic SUM((SEX=MALE) & (MAJOR=CS), GRADEPOINT) gives their total grade-point. Note that for $f\,(C,D)$ = COUNT(C), the set D is empty.

More general types of additive statistics can be expressed as finite moments over all M attributes [Dale82a]:

$$q\,(C) = \sum_{i \in C} x_{i1}^{e_1} x_{i2}^{e_2} \cdots\ x_{iM}^{e_M} , \tag{2.4}$$

where the exponents $e_1, \ldots, e_M$ are nonnegative integers. For COUNTs the exponents are all zero. For SUMs a single exponent is 1; all others are zero. In general, for $q\,(C) = f\,(C,D)$, the nonzero exponents correspond to the attributes in D.

Statistics derived from the values of d attributes are called **d-order statistics**. For additive statistics, the attributes can be specified by terms in the characteristic formula C, or by nonzero exponents e_j in Eq. (2.4). Specifically, a statistic $q\,(C)$ is of order d if C is an m-set and $d-m$ additional attributes have nonzero exponents in Eq. (2.4). For example, SUM((SEX=MALE) & (MAJOR=CS), GP)) is a 3-order statistic whose query set is a 2-set. COUNT(ALL) is a 0-order statistic.

It is customary to speak of **tables of statistics**. These correspond to tables of the lattice, where the cells of an m-table contain d-order statistics $f\,(C,D)$

for the m-sets C of the table. Census agencies, for example, typically release tables of COUNTs or SUMs for small m.

Property (2.3) implies that the additive statistics for the elementary sets in a table T_j can be computed from the statistics in a table T_i, where $T_i \leq T_j$. An additive statistic for an elementary set (A=a) & (B=b) of table AB in Figure 2, for example, can be computed from those for table ABC or table ABD. Using ABD, we have

$$q((A=a) \& (B=b)) = \sum_{d} q((A=a) \& (B=b) \& (D=d)) .$$

This result implies that to protect the values in a table T_j, it is necessary to protect statistics in the higher-dimensional tables T_i, where $T_i < T_j$ [Haq74a]. The converse, however, is not generally true; that is, it is not usually possible to compute statistics in higher-dimensional tables from those in lower-dimensional ones (exceptions to this rule are discussed in Section 6).

Figure 3 shows a table of student gradepoint sums. The entries inside the table belong to the 2-table over SEX and MAJOR. The row sums belong to the 1-table over SEX, and the column sums to the 1-table over MAJOR. The total 218.0 belongs to the 0-table ALL.

		MAJOR				
		ART	BOT	CS	EE	$\sum$
SEX	MALE	32.5	30.1	37.2	38.2	138.0
	FEMALE	18.5	3.9	31.3	26.3	80.0
	$\sum$	51.0	34.0	68.5	64.5	218.0

Figure 3. Gradepoint Sums by SEX and MAJOR.

Note that a d-order statistic over attributes $A_1, \ldots, A_d$ can be computed from the cardinalities (counts) of the elementary sets in the d-table defined by

$A_1, \ldots, A_d$. For example, suppose that the domain of the attribute GP is given by {0.00, 0.01, ..., 4.00}. Then the statistic SUM((SEX=FEMALE) & (MAJOR=CS), GRADEPOINT) is given by

$$\sum_{gp=0.00}^{4.00} \text{COUNT((SEX=MALE) \& (MAJOR=CS) \& (GRADEPOINT=}\boldsymbol{gp}\text{))} * \boldsymbol{gp} \ .$$

The lattice model outlined here is essentially that used by census agencies for disclosure analysis. A more complete description of the model is given in [Denn83a].

3. Disclosure of Sensitive Statistics.

A statistic is **sensitive** if confidential data could be deduced from the statistic alone. A statistic computed from confidential information in a query set of size 1 is always sensitive. A statistic computed from a query set of size 2 may also be classified as sensitive because a user with supplementary knowledge about one of the values can deduce the other from the statistic. The exact criterion for sensitivity is determined by the policies of the system. One criterion used by the U.S. Census Bureau for economic data is the "n-respondent, k%-dominance" criterion, which defines a sensitive statistic to be one where n or fewer records comprise more than k% of the total [Cox78a]; n and k are parameters of the database, usually kept secret. In this paper, we shall assume a sensitive statistic is one with a query set having fewer than n records.

Let R be a set of statistics released to some user. **Statistical disclosure** occurs when release of R allows the user to deduce something about a restricted statistic q. **Personal disclosure (compromise)** occurs when the user can deduce from R a sensitive statistic; that is, confidential information on an identifiable individual [Haq75a, Schl80a]. Disclosure may be either exact or approximate, positive or negative [Dale77a, Olss75a, Rapa75a].

Clearly, all sensitive statistics must be restricted (i.e., not permitted). In addition, it is necessary to restrict nonsensitive statistics that could lead to disclosure of sensitive ones.

Example . Suppose the formula C = (SEX = FEMALE) & (MAJOR = BOT) is known to identify Erna Weed. Then the statistic q = SUM((SEX = FEMALE) & (MAJOR = BOT), GRADEPOINT) shown in Figure 3 is sensitive and must be restricted. Moreover, additional statistics in the table must be restricted to prevent computing q; e.g., by

$$q = \mathrm{SUM}((\mathrm{MAJOR{=}BOT});\ \mathrm{GRADEPOINT}) - \mathrm{SUM}((\mathrm{SEX{=}MALE})\ \&\ (\mathrm{MAJOR{=}BOT});\ \mathrm{GRADEPOINT})\ . \qquad (3.1)$$

We shall now discuss memoryless cell-level controls, which protect sensitive statistics by suppressing all sensitive and some nonsensitive cells in a table. We turn to table-level controls, which suppress complete tables, in Section 5.

4. Cell Restriction Techniques

One of the first memoryless cell-level controls to be proposed was the **query-set-size control** [Fell72a, Hoff70a, Schl75a]. Given a query $q(C)$, the query-set-size control checks whether C and its "implied query set" $\sim C$ contain at least n records[+]; $\sim C$ is implied by C because $q(\sim C)$ can be computed from

$$q(\sim C) = q(\mathrm{ALL}) - q(C)\ .$$

Note that this control restricts $q(\mathrm{ALL})$. In practice, $q(\mathrm{ALL})$ cannot be hidden because it is easily computed using any permitted T by $q(\mathrm{ALL}) = q(T) + q(\sim T)$. We shall, therefore, assume $q(\mathrm{ALL})$ is released. Unfortunately, a query-set-size control is easily subverted, the most powerful methods being "trackers" [Denn79a, Denn80a, Schl75a, Schl80a, Schw77a].

We shall now investigate the feasibility of designing a secure memoryless cell-level control. This study was motivated by Friedman and Hoffman [Frie80a], who proposed to thwart tracker attacks by extending the concept of an implied query set to sets other than just complements.

Given a query $q(C)$ with m-set C over attributes $A_1, \ldots, A_m$, our objective is to determine a set I_C of m-sets over $A_1, \ldots, A_m$, called the **implied query sets** of C, such that for any $D \in I_C$, $q(D)$ can be computed from $q(C)$ plus other permitted statistics (these will be primarily lower-order statistics, which are

[+] In practice it is unnecessary to form the set $\sim C$ because $|\sim C| = N - |C|$; thus, we need only check that $n \leq |C| \leq N-n$.

less sensitive). An **implied queries control** restricts $q(C)$ if any set in I_C is sensitive (i.e., too small).

Note that the set I_C is limited to m-sets over one table, namely the m-table defined by $A_1, \ldots, A_m$ and associated with $q(C)$; $q(C)$ will correspond to either a single cell of the table or a union of cells . For a given query, we restrict attention to a single m-table because every sensitive statistic is associated with some m-table, and a sensitive statistic generally cannot be inferred without obtaining at least one statistic from its associated table. Statistics from lower-dimensional tables may be needed as well, but the objective is always to restrict at the highest level possible since lower-level statistics are more relevant to users. Thus, we prefer to protect a statistic in an m-table by suppressing other statistics for the table rather than lower-level statistics.

We first consider the case where C is expressed as an arbitrary formula; that is, the syntax is free or unrestricted (Section 4.1). Next we consider a partially restricted syntax, where formulas are constructed from clauses (Section 4.2). In both cases, the outcome is unsatisfactory. Finally, we restrict the syntax even further to logical AND (Section 4.3).

4.1. Free Syntax

Let C be an arbitrary formula defining an m-set over attributes $A_1, \ldots, A_m$. The following lemma and corollary show that for every elementary m-set E in the m-table $A_1 \cdots A_m$, there exists an m-set C' such that $q(E)$ can be computed from $q(C)$ and $q(C')$. This means that all s_m elementary sets in $A_1 \cdots A_m$ must be in I_C, because there is no way of predicting from $q(C)$ which query $q(C')$ will be asked and, since the control is memoryless, we cannot keep a record of those that are asked. Note, however, that if all elementary sets are included in I_C, then it is unnecessary to include any nonelementary m-sets, since these are simply unions of elementary sets (and therefore no more sensitive).

Lemma 1. Let C be an m-set over attributes $A_1,\ldots,A_m$, and let E be any elementary set in the m-table $A_1 \cdots A_m$. Then there exists an m-set C' such that either

$$E = C - C' , \quad \text{or} \quad E = C' - C . \tag{4.1}$$

Proof. Since C is a union of elementary sets in $A_1 \cdots A_m$, define C' by $C' = C - E$ if E is in C, and by $C' = C + E$ if E is not in C. ■

Corollary 1. Any additive statistic $q(E)$ can be computed from

$$q(E) = q(C) - q(C') , \quad \text{or} \quad q(E) = q(C') - q(C) . \tag{4.2}$$

Proof. Immediate from (4.1) and the additive property (2.3). ■

If indeed we must include all s_m elementary m-sets in an m-table in the set I_C for a given m-set C, then there are two serious problems with an implied queries control. First, the number of implied query sets that must be checked can become enormous even for small m if the domains of the attributes are large. For $m = 2$ and $|A_1| = |A_2| = 50$, for example, there are $s_m = 50 * 50 =$ 2,500 implied query sets for every 2-set C. Second, even though the control aims to restrict at the cell level, it effectively restricts entire m-tables. Given that we are going to restrict entire tables, we would like a more efficient method for deciding which tables to restrict (see Section 5).

4.2. Clause Syntax

We now consider the prospects of allowing partial tables using a restricted syntax. The clause syntax admits only a special class of characteristic formulas that define m-sets by the conjunction of m clauses. This syntax still allows a user to express most queries used in statistics production [Olss75a], including "range queries" such as

$$C = (20 \leq \text{AGE} \leq 30) \ \& \ (20\text{K} \leq \text{SALARY} \leq 30\text{K}) ,$$

where $(20 \leq \text{AGE} \leq 30)$ is shorthand for $(\text{AGE} = 20) + \ldots + (\text{AGE} = 30)$. Statistics Sweden has adopted the clause syntax [Olss75a, Rapa75a]. A preliminary report of the results of this subsection is given in [Denn81a].

A **clause** X is a formula of the form

$$X = (A = a_1) + (A = a_2) + \cdots + (A = a_k) , \tag{4.3}$$

where A is an attribute and each a_i $(1 \leq i \leq k)$ is a distinct value in the domain of A. Examples of clauses are

$$(\text{SEX}=\text{MALE}) ,$$
$$(\text{MAJOR}=\text{CS}) + (\text{MAJOR}=\text{EE}) + (\text{MAJOR}=\text{MATH}) .$$

An m-set C is defined by a conjunction*:

$$C = C_1 \ \& \ C_2 \ \& \ \cdots \ \& \ C_m , \tag{4.4}$$

where, for $1 \leq j \leq m$, $C_j \in \{X_j, \sim X_j\}$, and X_j is a clause over attribute A_j. The 3-set

$$(\text{SEX}=\text{MALE}) \ \& \ ((\text{MAJOR}=\text{CS})+(\text{MAJOR}=\text{EE})) \ \& \ (\text{CLASS}=1983) ,$$

for example, is formed from the conjunction of 3 clauses. Note that whereas formulas of the form (4.4) are syntactically restricted, they provide access to all additive statistics of the form (2.4) (assuming these statistics are released). This is because the formulas (2.2) defining the elementary sets are a special case of (4.4) where each clause names a single attribute value, and by (2.3), all additive statistics can be computed from those for the elementary sets.

Given a query set C of the form (4.4) over clauses $X_1, \ldots, X_m$, let $C^{\&}$ be the set of 2^m query sets of the conjunctive form (4.4), obtained by taking all

* Note that disjunctions of clauses $D = C_1 + C_2 + \cdots + C_m$ might be allowed as well. The complement $\sim D = \text{ALL} - D = \sim C_1 \ \& \ \sim C_2 \ \& \ \cdots \ \& \ \sim C_m$ conforms to (4.4). $\sim D$ is certainly an implied query set of D, and vice versa, so disjunctions of clauses have the same implied query sets.

combinations of each clause X_j or its complement $\sim X_j$. The following lemma shows that given any statistic $q(C)$ where C is of the form (4.4), then all 2^m statistics $q(D)$ for D in $C^{\&}$ can be computed from $q(C)$ using at most m lower-order statistics over subsets of $X_1, \ldots, X_m$. Because the lower-order statistics may be released, all sets in $C^{\&}$ must be included in the implied queries set I_C.

Lemma 2. Given $q_1 = q(C) = q(C_1 \,\&\, \cdots \,\&\, C_m)$, where C is an m-set of the form (4.4), let $q_2 = q(D) = q(D_1 \,\&\, \cdots \,\&\, D_m)$, where D is any m-set in $C^{\&}$. If q_1 and q_2 differ by k terms (i.e., $D_j = \sim C_j$ for k of the D_j), then q_2 can be expressed as a linear combination of q_1 and k lower-order statistics.

Proof. After replacing C_j by 1, $\sim C_j$ by 0 $(1 \le j \le m)$, apply the argument given in the proof of Theorem 3 (if-part) in Kam and Ullman [Kam77a].

Thus, with the clause syntax, an implied queries control must check at least the 2^m query sets in $C^{\&}$. Unfortunately, this is still insecure. To see why, consider Figure 4, which shows counts for a 2-table over attributes A and B. We write a_j for $A=a_j$ and b_j for $B=b_j$. The count $q(a_1 \,\&\, b_1) = 1$ is sensitive. Assuming an implied queries control checks only those sets in $C^{\&}$, with a permitted count of $n = 3$, it is easily verified that the control will permit $q(a_2 \,\&\, b_1) = 4$, $q(a_3 \,\&\, b_1) = 5$, and $q(b_1) = 10$. Thus, a user can compute

$$\begin{aligned} q(a_1 \,\&\, b_1) &= q(b_1) - q(a_2 \,\&\, b_1) - q(a_3 \,\&\, b_1) \\ &= 10 - 4 - 5 = 1\,. \end{aligned}$$

This example is easily generalized to m-tables with $m > 2$.

		B		
		b_1	b_2	b_3
	a_1	1	3	6
A	a_2	4	7	5
	a_3	5	3	6

Figure 4. Cardinalities for 2-table over attributes A and B.

Experience with a host of tables from real databases [Schl82a] shows that such attacks are frequently possible. It is at best questionable whether a memoryless control can prevent such attacks without inspecting many, if not all, elementary m-sets in an m-table, and whether such a control can provide security without suppressing entire tables. With the clause syntax, an implied queries control faces similar difficulties as with a free syntax, its cost-benefit ratio is unsatisfactory, and it is probably impractical.

4.3. AND Syntax

We consider query sets restricted to logical AND, disallowing OR and NOT. Thus, every formula is of the form (2.2) and defines an elementary m-set ($0 \leq m \leq M$). Writing a_{ij} for $A = a_{ij}$, we get formulas of the general form

$$E = a_{i_1} \; \& \; a_{i_2} \; \& \; \cdots \; \& \; a_{i_m} \, . \qquad (4.5)$$

An additive statistic $q(E)$ with query set E of the form (4.5) corresponds to a cell of the m table defined by $A_1,\ldots,A_m$.

Applying previous results [Cox80a, Kam77a, Olss75a, Schl76a, Yao79a], we first observe that to protect a sensitive cell $q(E)$, at least 2^m cells $q(E')$ must be suppressed, where the elementary sets E' fall into a hypercube that includes E. Moreover, for counts the suppressed cube may contain zeros only in certain positions [Olss75a, Kam77a]. A 2^m-cube around a sensitive cell $q(E)$ is defined by

$$cube(E) = \{E' \mid E' = a_1^{\bullet} \;\&\; \cdots \;\&\; a_m^{\bullet}\}, \qquad (4.6)$$
$$a_j^{\bullet} \in \{a_{i_j}, a'_{i_j}\},$$

where a_{i_j} and a'_{i_j} are distinct values in the domain of attribute A_j. Given a sensitive cell $q(E)$, the problem is to choose the partner+ a'_{i_j} for a_{i_j} $(1 \le j \le m)$. For an attribute A_j with $|A_j|$ values, there are $|A_j| - 1$ ways to choose a partner. Once the partners have been selected, they must remain constant for all future queries.

The general approach of restricting hypercubes is fundamental to the cell suppression controls used by census agencies [Cox78a, Cox80a, Olss75a, Rapa75a, Sand77a]. These controls analyze the linear relationships among all cells of a table (including the marginal sums) to determine whether sensitive cells can be estimated too closely from those that are released; additional cells are suppressed until this is no longer possible. Some attempt is made in selecting hypercubes for suppression to minimize information loss. With current technology, this linear analysis approach to cell suppression would be very expensive to apply as a memoryless control on a per query basis. Although the analysis time could be reduced by periodically computing and storing a complete set of tables for statistical purposes (e.g., by using the "box structures" employed by Statistics Sweden [Sund78a]), this is not likely to be practical for multipurpose database systems in the near future.

We therefore investigated a heuristic approach to cell suppression [Wehr83 , Wehr84]. Assuming the values in each attribute domain are cyclically ordered (modulo the size of the domain), we introduce, as a parameter of the database, an integer vector

+ The partner relation need not be symmetric.

$$X = (x_1, x_2, \ldots, x_M) .$$

The index of the partner of a_{i_j} is then given by

$$i_j' = [(i_j + x_j - 1) \bmod |A_j|] + 1 .$$

For $X = (1, -1)$, for example, the partners for a_1 and b_1 in Figure 4 are a_2 and b_3, respectively; the 2^2-cube for the sensitive cell $E = q(a_1 \& b_1)$ in Figure 4 will thus be given by

$$cube(E) = \{a_1\&b_1 , a_1\&b_3 , a_2\&b_1 , a_2\&b_3\} .$$

In practice, the control works backwards: Using X, for each incoming query $q(E')$, the system determines if there exists a sensitive cell $q(E)$ whose hypercube contains $q(E')$, in which case $q(E')$ is suppressed. In the worst case, this requires testing 2^m cells.

The heuristic does not guarantee security. Figure 5 shows how a suppressed 2-cube can be deduced when it is embedded in a larger 2-table that extends below and to the right of the cube. All other entries in the 2-table must be permitted along with the 1-tables so that the marginal sums for the cube can be deduced. Because the cells that border the cube are thus permitted, the unknowns x_{12}, x_{21}, and x_{22} must have met the sensitivity threshold, whence they must all be n and x_{11} must be 1. The interested reader may wish to verify that a 2-cube with values $(1, n+1, n, n)$ is also deducible.

$$\begin{array}{c|c|c|c} & n+1 & 2n & \\ \hline n+1 & x_{11} & x_{12} & \cdots \\ \hline 2n & x_{21} & x_{22} & \cdots \\ \hline & \vdots & \vdots & \end{array} \quad \rightarrow \quad \begin{array}{c|c|c|c} & n+1 & 2n & \\ \hline n+1 & 1 & n & \cdots \\ \hline 2n & n & n & \cdots \\ \hline & \vdots & \vdots & \end{array}$$

Figure 5. Deducible Hypercube.

Details of the research are described elsewhere [Wehr83 , Wehr84]. Here we mention only a few results. While not perfectly secure, the control can provide a high degree of security. It offers the possibility of allowing partial tables, but does not necessarily minimize information loss in those tables. The procedure can be extended to handle clauses of the form (4.4) (Section 4.2) in such a way that the average number of tests per elementary set contained in a query set decreases. If an entire table of statistics is produced at once, the overhead decreases further.

Whether the overhead of testin gup to 2^m cells per requested cell is acceptable will depend on the size of the database, the speed of the query processor, and the load on the system. If 2^m-cells can be quickly checked without degrading system performance, the control may be acceptable. Otherwise, it may be preferable to apply table-level controls, which are potentially much more efficient since they aim to restrict entire tables. We now investigate criteria for restricting complete tables.

5. Table Restriction Techniques

We now turn to controls that suppress complete tables of statistics. We assume, however, that queries are still for individual statistics corresponding to cells or cell unions (e.g., as with range queries), and that the syntax is unrestricted. Thus, we shall formulate our controls in terms of how they respond to such queries.

We initially consider queries for counts; that is, $q(C) = \text{COUNT}(C) = |C|$. Thus each m-table T over attributes $A_1, \ldots, A_m$ gives the cardinalities for the $s_m = \prod_{j=1}^{m} |A_j|$ m-sets over $A_1, \ldots, A_m$. If there are M possible attributes, then there are $\binom{M}{m}$ possible m-tables. We shall write T^m to denote any one of these tables.

Given a table T^m, let I_m be the number of m-sets in T^m of cardinality 1; I_m is the number of **identifications** in T^m. The relative **identification risk** of the table is defined by $p_m = I_m / N$, where N is the number of records in the database.

Now, if m attributes are needed to identify some individual, then $m+1$ attributes are generally needed to retrieve unknown values of the individual using counts [Fell72a, Fell74a, Hoff70a]. (There are exceptions to this principle, but we defer these to Section 6.) As an example, suppose Erna Weed is known to be the only female botany major represented in the student database. Although she is uniquely identified in the 2-table defined by SEX and MAJOR (see Figure 1), the statistic |(SEX=FEMALE) & (MAJOR=BOT)| = 1 does not disclose new information about her. Statistics released over the 3-table defined by SEX, MAJOR, and GRADEPOINT, on the other hand, could reveal her grade-point. This leads to the following rule:

> The **m+1 - rule**: For every m-table T^m such that $I_m \geq 1$, restrict every table T^k such that $T^k < T^m$; that is, such that $k \geq m+1$ and T^k is a descendent (refinement) of T^m in the lattice of tables. ▪

Note that this rule restricts the descendents of tables having nonzero identifications but not the tables themselves (unless they in turn are descendents of tables having nonzero identifications).

The $m+1$-rule is impractical as a query restriction control because it requires inspecting the size of every m-set in a table T^m. But it provides a yardstick for evaluating the effectiveness of more efficient techniques.

We shall now describe four such techniques: order, relative table size (s_m / N-criterion), minimum frequency ($\Pi\, rmin$-criterion), and explicit risk estimation.

Order. Given an m-set C, $q(C)$ is allowed only if $m \leq d$, where d is a parameter of the database. Thus, an m-table is restricted if $m > d$. •

Because the order restriction treats all m-tables uniformly, it can restrict many tables that do not threaten privacy in order to protect those that do. This is especially true if d is chosen to satisfy the $m+1$-rule for all tables. For example, let T_1^3 and T_2^3 be two 3-dimensional tables of counts in a database with $N = 1200$ records, where the records are uniformly distributed over all elementary sets. Suppose that each attribute defining T_1^3 has 2 values, and that each attribute defining T_2^3 has 10 values. Now, table T_1^3 has a total of 8 cells (elementary sets), so the average number of records in each cell is 1200/8 = 150. Thus, it is likely that enough records fall into each cell that no cell is sensitive. Table T_2^3, on the other hand, has 1000 cells, so the average number of records in each cell is only 1200/10 = 1.2. Thus, most cells will be sensitive, and T_2^3 must be suppressed. Choosing $d = 2$ protects T_2^3, but unnecessarily causes T_1^3 to be restricted as well.

In general, statistics computed over attributes with many values are more likely to be sensitive than those computed over the same number of attributes but with only a few values. The relative table size restriction decides which tables to release on the basis of the relative number of elementary sets in the table. The control is as follows:

Relative table size (s_m/N-criterion). Given an m-set C over attributes $A_1, \ldots, A_m$, $q(C)$ is permitted only if

$$s_m / N = \prod_{i=1}^{m} |A_i| / N \leq 1/k \; , \qquad (5.1)$$

where k is a parameter of the database. Thus, an m-table is restricted if its relative size s_m/N exceeds $1/k$. •

Rewriting (5.1), this says that $q(C)$ is permitted only if $N/s_m \geq k$; that is, each of the s_m cells in the table contains, on the average, at least k records. If k = 10, for example, each cell must contain, on the average, at least 10 records; thus, T_1^3, but not T_2^3, would be permitted in the above example. The control is particularly simple to implement; the system need only know N and the sizes of the attribute domains.

The criterion goes back to an observation by Block and Olsson [Bloc76a] that the identification risk in a table T^m will be approximately

$$p_m = e^{-N/s_m} \tag{5.2}$$

if all m attributes in the table are independent and equidistributed. In real databases, the attributes are more or less interdependent and/or nonuniformly distributed. To determine the practical role of s_m/N, we inspected pairs of values (s_m, p_m) in 27 real databases or subdatabases. For each such pair, we computed (x,y)-coordinates defined by

$$\begin{aligned} x &= \log_{10}(s_m/N) \\ y &= -\log_{10}(-\ln(p_m)) . \end{aligned}$$

Now, if (5.2) holds, then

$$y = -\log_{10}(-\ln(e^{-N/s_m})) = \log_{10}(s_m/N) = x ,$$

whence the (x,y) points will fall in a straight line. Plotting the (x,y) points for the 27 databases showed that the points for each database fit fairly well a straight line, called the **risk line**. Figure 6 shows the points for Database 12. Figure 7 shows the risk lines for all databases calculated from all observed risk points with $0.00316 \lesssim s_m/N \leq 1$ $(-2.5 \lesssim \log_{10}(s_m/N) \leq 0)$. In this region, the points fit a straight line closest for most observed databases. The slopes of the risk lines are lower than that predicted by Eq. (5.2) because the attributes are not all independent and equidistributed. The slopes increase as the attributes approach equidistribution and have weaker interdependencies. Although the risk lines were somewhat different for each database, all intersected the line for

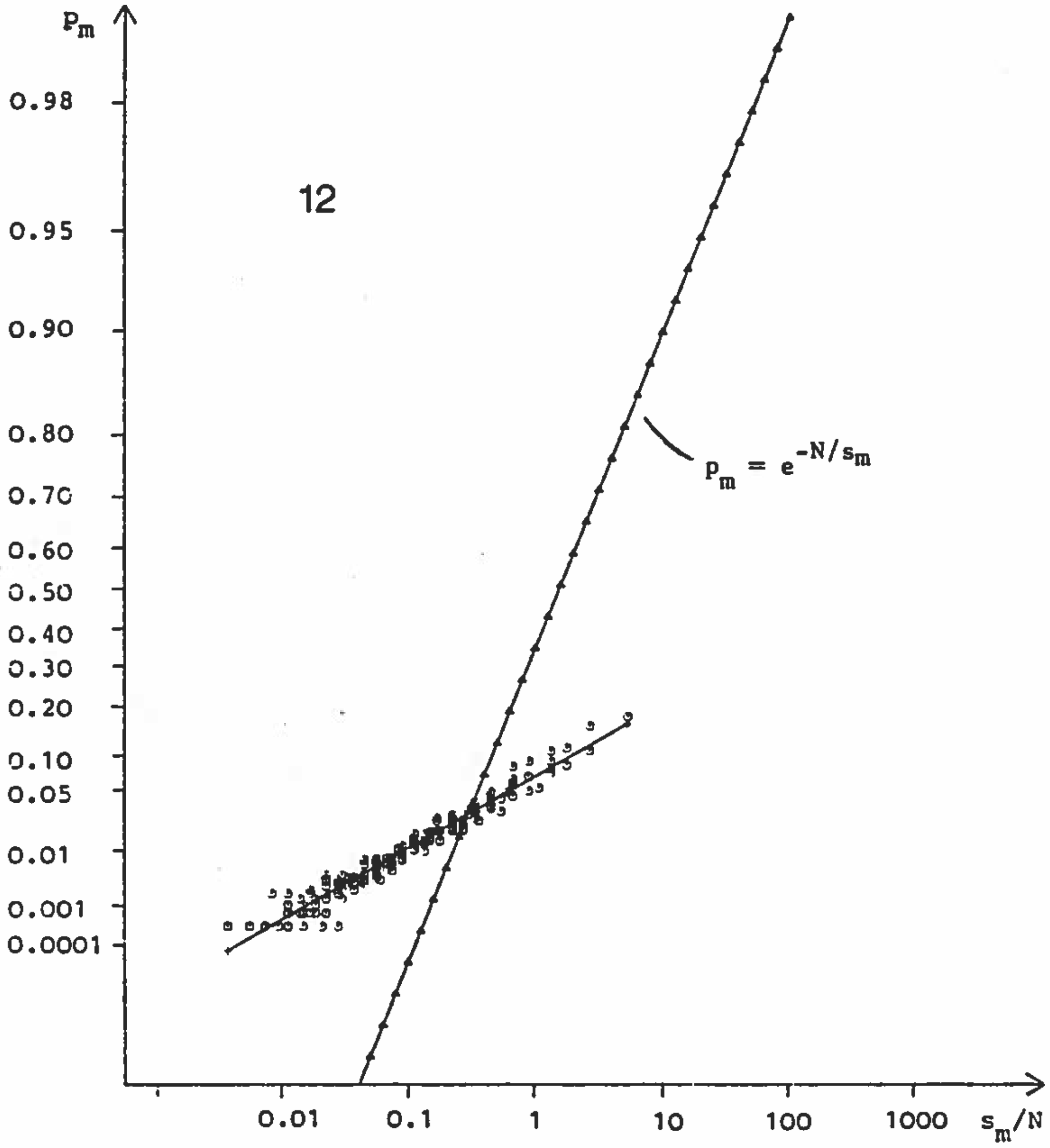

Figure 6. Risk line for Database 12.

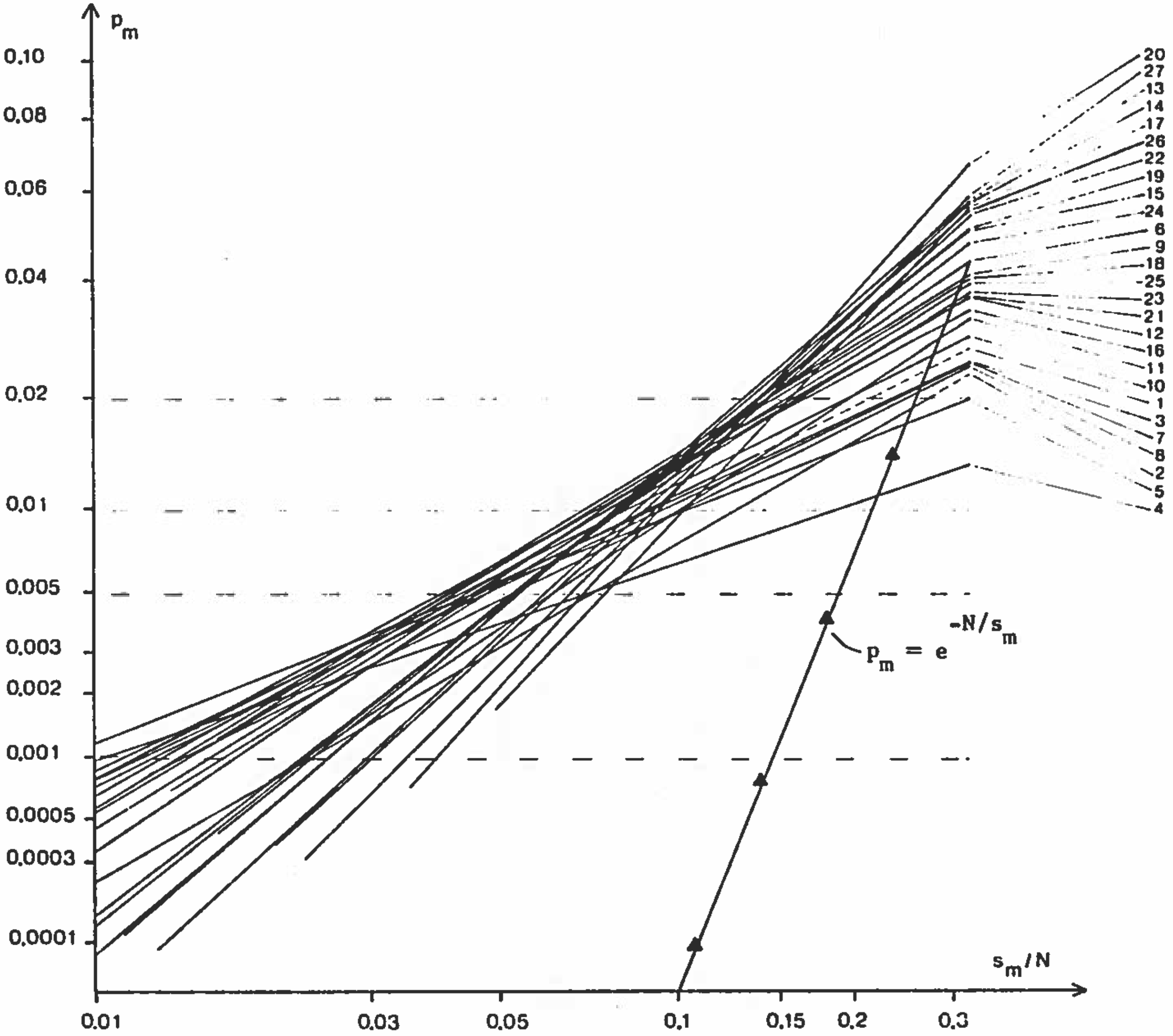

Figure 7. Risk lines for 27 databases computed from all observed risk points with $s_m/N \leq 1$.

Eq. (5.2) at a point corresponding to a value of s_m / N somewhere between 0.22 and 0.42. For $s_m / N \lesssim 0.3$, the identification risk p_m is higher than predicted by (5.2); for $s_m / N \gtrsim 0.3$, it is lower. Even more interesting, we found that for s_m / N around 0.1, all 27 investigated databases have fairly similar identification risks. (See [Schl82a] for a detailed report of these results.) The database system used in this study evolved from one designed by Selbmann [Selb74a].

Because the s_m / N-criterion is based only on relative table size, it cannot recognize tables with identifications when the average cell size is k or more. The third and fourth controls aim to recognize such tables by using frequency distributions. The third control, first proposed in [Schl76a], uses minimum relative frequencies:

Minimum frequency ($\Pi\, rmin$-criterion). Given an m-set C over attributes $A_1, \ldots, A_m$, $q(C)$ is permitted only if

$$\Pi\, rmin = \prod_{j=1}^{m} min(r_{A_j}) \geq k / N , \tag{5.3}$$

where $min(r_{A_j})$ is the smallest relative frequency occurring among the values in the domain of A_j, and k is a parameter of the database. Thus, an m-table is restricted if $\Pi\, rmin < k / N$. ▪

Because it uses only the minimum relative frequency of each attribute, $\Pi\, rmin$ is able to predict first but not subsequent identifications in a table. Our fourth control uses complete frequency distributions to explicitly estimate the identification risk.

Explicit risk estimation (from parents). Let C be an $(m+1)$-set on table T^{m+1}, and let $T_1^m, \ldots, T_{m+1}^m$ be the parents of T_{m+1} in the lattice; i.e., $T^{m+1} < T_j^m$ $(1 \leq j \leq m+1)$. Then $q(C)$ is permitted only if for $j = 1, \ldots, m+1$, $\hat{I}_j < z$, where $\hat{I}_j$ is an estimate of the number of identifications in table T_j^m, and z is a parameter of the database. Thus, all descendents of tables having z or more estimated identifications are restricted. Each $\hat{I}_j$ is computed using frequency distributions over the m attributes in T_j^m. •

Note that for $z = 1$, the control attempts to approximate the $m+1$-rule. If the estimators $\hat{I}$ are close to the actual identifications, then the approximation should be very good. We shall also consider a simplified version of the control which uses the estimator $\hat{I}$ associated with the table itself rather than those of its parents:

Explicit risk estimation (from table). Given an m-set C on table T^m, $q(C)$ is permitted only if $\hat{I} < z$, where $\hat{I}$ is an estimate of the number of identifications in table T^m. Thus, all tables with z or more estimated identifications are restricted. •

For an m-table T^m over attributes $A_1, \ldots, A_m$, we shall base our estimator $\hat{I}$ on the 1-dimensional relative frequency distributions of $A_1, \ldots, A_m$. For each attribute A_i $(1 \le i \le m)$, let $r_{ij} = |(A_i = a_{ij})| / N$ be the relative frequency of value a_{ij} $(1 \le j \le |A_i|)$. Define each of the s_m elementary sets in T^m by

$$E_t = (A_1 = a_{1t_1}) \;\&\; \cdots \;\&\; (A_m = a_{mt_m}) \,.$$

For each elementary set E_t $(1 \le t \le s_m)$, the relative frequency $|E_t| / N$ is estimated by

$$\hat{r}_t = r_{1t_1} \;\&\; \cdots \;\&\; r_{mt_m} \,,$$

and the probability E_t contains exactly one record by

$$\hat{p}_t \;=\; \text{prob}[\,|E_t| = 1\,] \;=\; \binom{N}{1} \hat{r}_t^{\,1} \quad (1-\hat{r}_t)^{N-1} .$$

The identification risk for T_m is then estimated by

$$\hat{I} = \sum_{t=1}^{s_m} \hat{p}_t \,. \tag{5.4}$$

(The computation of $\hat{I}$ is optimized as described for Algorithm 1 in [Schl82a].)

To evaluate these controls, we chose 9 of the 27 databases of Figure 7 for further study. Table 1 describes these databases, plus an additional one derived from Database 11 (referred to as Database 11'). The databases are ordered by increasing s_u / N. Database 11 is a subset of Database 1, containing only 606 of the 31,075 records in Database 1: since only 2 of the 4 values for attribute E appear in Database 11, $|E| = 2$ in Database 11, whereas $|E| = 4$ in Database 1. Comparisons between Databases 11 and 1 illustrate the effect of increasing the database size N while holding the number of attributes (and thereby the number of tables) fixed. Database 11' was derived from Database 11 by deleting attribute E (attribute F in Database 11 was renamed E in Database 11'). Comparisons between Databases 11' and 11 illustrate the effect of increasing the number of attributes or tables (and thereby the ratio s_u / N) for fixed N. Most of these databases have relatively flat risk lines; that is, their identification risks are relatively high in the region $s_m / N \le 0.1$, which we are mainly interested in here. For comparison, Database 13 with a steep risk line is included.

Database 11' is small enough for a detailed presentation of each table in the lattice. Figure 8 depicts the complete lattice, showing s_m / N and the identification risk for each table. The s_m / N ratios range from 0.00 to 0.01 for the 1-tables, from 0.01 to 0.07 for the 2-tables, from 0.04 to 0.30 for the 3-tables, and from 0.20 to 0.89 for the 4-tables; the ratio for the 5-table is 1.78. Picking $k = 10$ with the s_m / N criterion, for example, would restrict none of the 1- and 2-tables, 5 (out of 10) 3-tables, all 5 4-tables, and the 5-table. The identification risks for these tables range from 1.0 percent to 10.6 percent. The risks for the permitted tables, on the other hand, range from 0 to 1.2 percent.

Table 2 shows the number of permitted and restricted tables, and the number and percent of accessible (compromisable) data values for each of the controls applied to the databases of Table 1. For comparison, the number of

N = number of records
M = number of attributes
NM = number of data elements
$\#T$ = number of tables (including the 0-table)
$|A|$ = number of values of attribute A
s_M = number of elementary sets = $|A| \bullet |B| \bullet \ldots$

Database	N	M	MN	$\#T$	|A|	|B|	|C|	|D|	|E|	|F|	|G|	|H|	|I|	s_M	s_M / N
1	31075	6	186450	64	2	9	5	4	4	3				4320	0.14
8	2052	5	10260	32	5	5	6	4	5					3000	1.46
9	3259	6	19554	64	3	2	3	6	7	7				5292	1.62
11'	606	5	3030	32	2	9	5	4	3					1080	1.78
11	606	6	3636	64	2	9	5	4	2	3				2160	3.56
12	2152	8	17216	256	2	2	3	4	3	4	4	5		11520	5.35
13	15911	6	95466	64	12	6	6	10	6	5				129600	8.15
16	31465	8	251720	256	7	4	6	7	9	7	5	6		2222640	70.64
18	736	8	5888	256	3	4	4	5	4	4	6	6		138240	187.8
23	736	9	6624	512	2	4	8	3	4	5	6	6	8	1105920	1502.6

Table 1. Descriptions of databases.

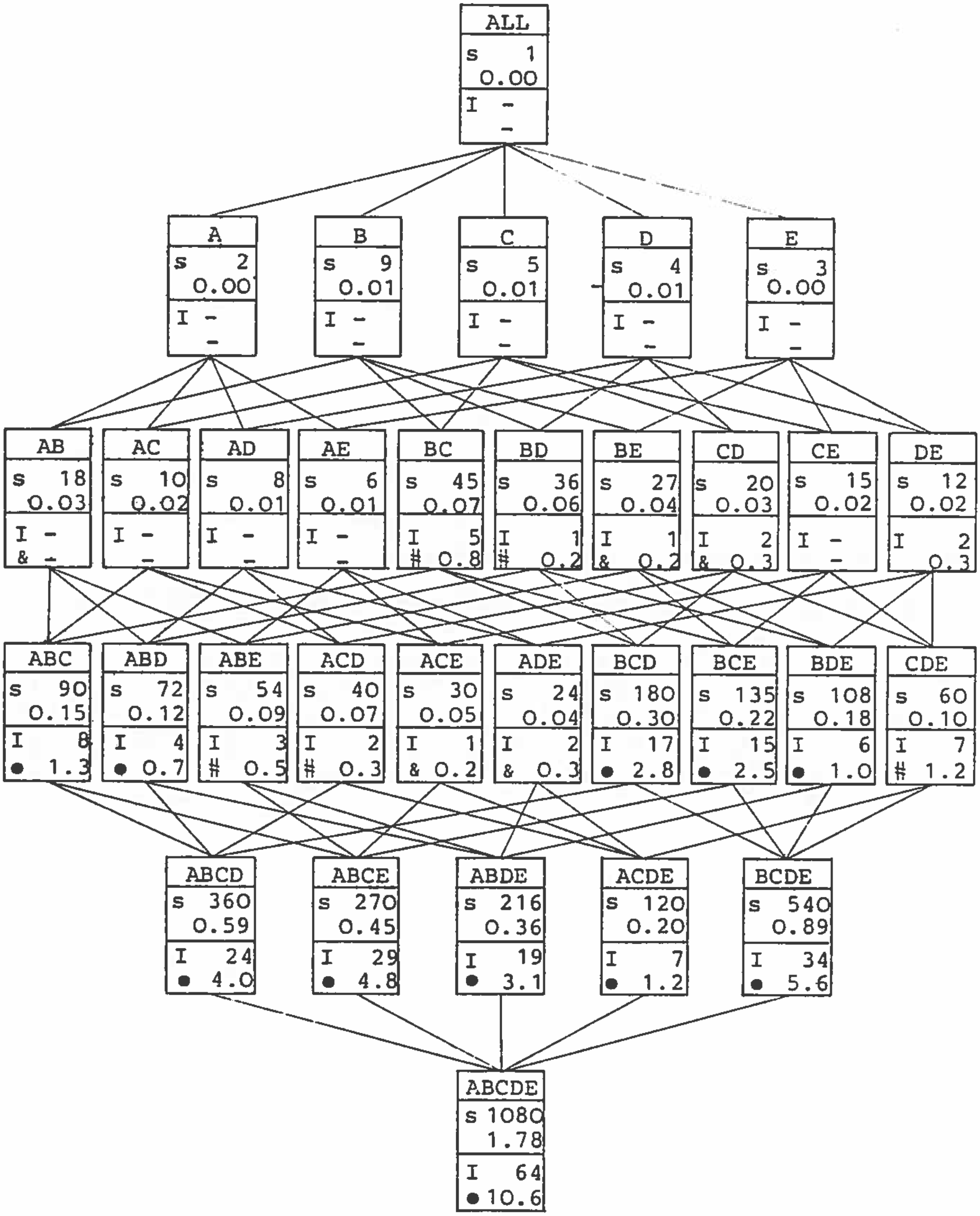

Figure 8. Lattice of tables for Database 11'. For each table T^m, $s = s_m$ = #elementary m-sets; the number below s is the quotient $s_m / N = s_m / 606$. $I = I_m$ = #identifications; the number below I is the identification risk in percent; i.e., $100p_m = 100I / 606$. Symbols: &: $0.025 \le s_m / N < 0.05$; #: $0.05 \le s_m / N < 0.10$; ●: $0.10 \le s_m / N$.

tables permitted and restricted by the $m+1$-rule are also given. For each control, the number of falsely permitted tables and percent of accessible data values measures its disclosure risk. Note that a large number of falsely permitted tables does not always imply a high percent of accessible data values. The number of falsely restricted tables measures the control's information loss relative to the $m+1$-rule. Table 3 summarizes the data in Table 2.

To determine the percent of accessible data values, we counted the number of data values accessible (by solving linear equations) in each of the databases under each of the controls. Table 4 shows how this is done for Database 11'. For each permitted table T^m, we count the number I_m of identifications (sets with cardinality 1) in T^m. For instance, table CD has 2 identifications, whereas table AB has none. For each T^m having $I_m > 0$, we determine which values can be retrieved for the individuals (records) identified in T^m from the permitted descendents T^{m+1} of T^m. For instance, the values for attributes A and E can be retrieved for the 2 records identified in table CD because both ACD and ACE are permitted. Nothing can be retrieved for the 5 records identified in table BC because all descendents of BC are restricted. The procedure does not count duplicate retrievals. Suppose, for example, that AB, ABC, ABD, and ABCD are permitted, where $I_2 = 1$ for table AB. For the identified record, the values of C and D may be retrieved using ABC and ABD. Because the record is also identified in table ABC (and ABD), the values of D (or C) can also be retrieved using ABCD. The procedure counts the accessible values of C and D only once. Even then, the procedure only gives an upper bound on the number of retrievable values because it does not check record identifiers to determine whether a record identified in one table is the same as that identified in another table. For instance, if one of the records identified in CD is also identified in DE, then the value retrieved for A in ACD is the same as that retrieved in ADE.

Restriction criterion	Database	perm	rest	fp	fr	acc	%
	1	41	23	-	-	-	-
$m+1$-rule:	8	16	16	-	-	-	-
	9	30	34	-	-	-	-
all descendents of	11'	17	15	-	-	-	-
tables with at least	11	29	35	-	-	-	-
one identification	12	69	187	-	-	-	-
restricted	13	38	26	-	-	-	-
	16	62	194	-	-	-	-
	16	39	217	-	-	-	-
	23	49	463	-	-	-	-
	1	22	42	-	19	-	-
Order, $d = 2$:	8	16	16	-	-	-	-
	9	22	42	-	8	-	-
m-tables with $m > 2$	11'	16	16	-	1	-	-
restricted	11	22	42	-	7	-	-
	12	37	219	-	32	-	-
	13	22	42	-	16	-	-
	16	37	219	-	25	-	-
	18	37	219	-	2	-	-
	23	46	466	-	3	-	-
	1	42	22	4	3	4	0.00
Order, $d = 3$:	8	26	6	10	-	30	0.29
	9	42	22	12	-	32	0.16
m-tables with $m > 3$	11'	26	6	9	-	33	1.09
restricted	11	42	22	14	1	44	1.21
	12	93	163	24	-	30	0.17
	13	42	22	4	-	4	0.00
	16	93	163	31	-	48	0.02
	18	93	163	54	-	288	4.89
	23	130	382	81	-	616	9.30
	1	58	6	17	-	172	0.09
Relative table size,	8	16	16	-	-	-	-
$k = 40$	9	32	32	4	2	6	0.03
	11'	11	21	-	6	-	-
tables with $s_m / N > 1/40$	11	17	47	-	12	-	-
restricted	12	90	166	25	4	56	0.33
	13	35	29	3	6	3	0.00
	16	94	162	32	-	64	0.03
	18	22	234	-	17	-	-
	23	24	488	-	25	-	-
	1	62	2	21	-	454	0.24
Relative table size,	8	19	13	3	-	5	0.06
$k = 20$	9	44	20	14	-	46	0.24
	11'	16	16	1	2	2	0.07
tables with $s_m / N > 1/20$	11	27	37	2	4	4	0.11
restricted	12	128	128	59	-	265	1.54
	13	42	22	4	-	4	0.00
	16	134	122	72	-	1890	0.75
	18	37	219	-	2	-	-
	23	45	467	4	8	11	0.17
	1	63	1	22	-	615	0.33
Relative table size,	8	26	6	10	-	30	0.29
$k = 10$	9	51	13	21	-	172	0.88
	11'	21	11	4	-	9	0.30
tables with $s_m / N > 1/10$	11	38	26	9	-	20	0.55
restricted	12	165	91	96	-	799	4.64
	13	43	21	5	-	22	0.02
	16	163	93	101	-	4925	1.96
	18	59	197	20	-	59	1.00
	23	74	438	25	-	86	1.30

Table 2. Number of permitted and restricted tables, and number and percent of accessible (compromisable) data values.

Restriction criterion	Database	perm	rest	fp	fr	acc	%
Minimum frequency, $k = 10$ tables with $\Pi rmin < 10/N$ restricted	1	41	23	2	2	2	0.00
	8	6	26	-	10	-	-
	9	8	56	-	22	-	-
	11'	9	23	-	6	-	-
	11	12	52	-	17	-	-
	12	22	234	-	47	-	-
	13	18	46	-	20	-	-
	16	24	232	-	38	-	-
	18	9	247	-	30	-	-
	23	10	502	-	39	-	-
Minimum frequency, $k = 1$ tables with $\Pi rmin < 1/N$ restricted	1	54	10	13	-	76	0.04
	8	16	16	-	-	-	-
	9	18	46	-	12	-	-
	11'	18	14	3	2	5	0.17
	11	29	35	4	4	6	0.17
	12	53	203	-	16	-	-
	13	28	36	-	10	-	-
	16	49	207	1	14	1	0.00
	18	26	230	-	13	-	-
	23	25	487	1	25	2	0.03
Explicit risk estimation from parents, $z = 0.5$: descendents of tables with ≥ 0.5 estimated identifications restricted	1	57	7	16	-	121	0.06
	8	20	12	4	-	10	0.10
	9	28	36	3	5	5	0.03
	11'	18	14	2	1	3	0.10
	11	30	34	4	3	6	0.17
	12	69	187	6	6	6	0.03
	13	42	22	4	-	4	0.00
	16	73	183	13	2	14	0.01
	18	47	209	8	-	18	0.31
	23	49	463	2	2	4	0.06
Explicit risk estimation from parents, $z = 1.0$: descendents of tables with ≥ 1.0 estimated identifications restricted	1	57	7	16	-	121	0.06
	8	26	6	10	-	30	0.29
	9	35	29	5	-	8	0.04
	11'	19	13	3	1	5	0.17
	11	32	32	6	3	14	0.39
	12	95	161	26	-	40	0.23
	13	42	22	4	-	4	0.00
	16	87	169	27	2	41	0.02
	18	50	206	11	-	26	0.44
	23	54	458	6	1	15	0.23
Explicit risk estimation from table, $z = 2.0$: tables with ≥ 2.0 estimated identifications restricted	1	51	13	10	-	37	0.02
	8	16	16	-	-	-	-
	9	22	42	-	8	-	-
	11'	17	15	2	2	3	0.10
	11	28	36	3	4	5	0.14
	12	64	102	5	10	4	0.02
	13	24	40	-	14	-	-
	16	43	213	-	19	-	-
	18	30	226	-	9	-	-
	23	31	481	1	19	2	0.03

perm	=	# of permitted tables
rest	=	# of restricted tables
fp	=	# of falsely permitted tables (by m+1-rule)
fr	=	# of falsely restricted tables (by m+1-rule)
acc	=	# of accessible data values
%	=	percent of accessible data values (acc/MN)

Table 2. Number of permitted and restricted tables, and number and percent of accessible (compromisable) data values. (continued)

$27t_3$

Criterion	fp	fr	fp + fr	Average % of accessible data values
m + 1	-	-	-	-
Order, d=2	-	113	113	-
Order, d=3	243	4	247	1.71
s_m / N, k=40	81	72	153	0.05
s_m / N, k=20	180	16	196	0.32
s_m / N, k=10	313	-	313	1.13
$\Pi rmin$, k=10	2	233	235	0.00
$\Pi rmin$, k=1	22	96	118	0.04
Exp. risk est., parent, z=0.5	62	19	81	0.09
Exp. risk est., parent, z=1.0	114	7	121	0.19
Exp. risk est., table, z=2.0	21	85	106	0.03

Table 3. Summary of data in Table 2.

$27t_4$

m	Table	s_m	I_m	Access
0	ALL	1	-	
1	A	2	-	
1	B	9	-	
1	C	5	-	
1	D	4	-	
1	E	3	-	
2	AB	18	-	
2	AC	10	-	
2	AD	8	-	
2	AE	6	-	
2	BC	45	5	--
2	BD	36	1	--
2	BE	27	1	A - 1
2	CD	20	2	AE - 2
2	CE	15	-	
2	DE	12	2	AC - 2
3	ABE	54	3	--
3	ACD	40	2	--
3	ACE	30	1	--
3	ADE	24	2	--
3	CDE	60	7	--

Table 4. Data values accessible under the s_m / N-criterion with k = 10 for Database 11'. Only the permitted tables are listed (see also Figure 3).

Looking at the results for Databases 1 and 11 in Table 2, we observe that for all controls except order with $d = 3$, the number of accessible data values and false permissions is higher for Database 1 than for its record subset Database 11; accordingly, Database 1 exhibits fewer false restrictions. On the other hand, the percent of accessible data values is often even lower for Database 1 than for Database 11. A similar picture emerges from a comparison between Databases 16 (N = 31,465) and 12 (N = 2,152), which both have M = 8 and very similar risk lines. For most criteria, Database 16 shows more accessible data values and more false permissions than its low-N twin Database 12, while the percent of accessible values is usually lower for Database 16. This suggests that the relative risk, if anything, drops with increasing N, while the absolute number of data values at risk grows, at least with restriction criteria like relative table size or minimum frequency that directly depend on N. This is consistent with our finding from 27 databases that, other things being equal, increasing N diminishes the relative identification risk p_m, even though the quotient s_m / N corresponding to the first identification decreases [Schl82a]. The early identifications that arise with increasing N correspond to a discontinuity phenomenon known as "outliers" (records with extreme or otherwise unusual value combinations; e.g., see [U.S.78a]).

We also observe that for all controls, the percent of accessible values and the number of falsely restricted tables is at least as great for Database 11 as for Database 11'. This suggests that for fixed N, increasing the number of attributes (tables) increases the disclosure risk.

The risk for Database 13 is extremely low for all controls. This is explained in part by the steep risk line of the database.

We now make some observations about each of the controls.

Order. Table 2 shows that for the order restriction, picking $d = 2$ to satisfy the $m+1$-rule is too restrictive; increasing d to 3 is too permissive. We conclude that order is probably not useful as a stand-alone control, though it might be useful when combined with other controls.

Relative table size (s_m / N). Table 2 shows that the s_m / N-criterion can control disclosure without falsely restricting too many tables allowed by the $m+1$-rule. The most appropriate value for the parameter k, however, seems to depend on the database. Using the percent of accessible data values as a criterion in selecting k, $k = 10$ is appropriate for Database 13; $k = 20$ is appropriate for Databases 8, 11', 11, 18, and 23; $k = 40$ is appropriate for Databases 1, 9, and 16; and $k > 40$ may be needed for Database 12. Note that even for these values of k, some tables are falsely permitted. Yet if we try to find a k that does not permit any table restricted by the $m+1$-rule, we are led to much larger values of k. For Database 1, for example, we would need $k = 345$, which would lead to an excessive number of false restrictions. Moreover, from Table 2 we see that considerable information will be lost if we stick too closely to the $m+1$-rule, restricting the descendents of every table having a nonzero identification risk. A better strategy (in terms of minimizing information loss) might be to restrict only those tables with a high risk, and use other techniques to thwart attacks on the permitted tables. This approach is discussed in the next section.

For $k = 10$ and 20, we see that the databases having the greatest number of permitted tables seem to have the greatest percent of accessible values. This becomes even clearer if we exclude tables with a negligible risk. Table 5 shows that there is a correlation between the number of permitted tables with $1/10 \leq s_m / N \leq 1/k$ and the risk. ($k = 40$ is omitted from the analysis since the number of accessible values is low for all databases except Database 12. Since Database 12 has a much higher risk than the other databases, the correlations

Database	$k = 20$		$k = 10$	
	x	y	x	y
1	11	0.243	12	0.330
8	10	0.058	17	0.292
9	24	0.235	31	0.880
11	18	0.110	29	0.550
12	83	1.539	121	4.641
16	55	0.751	84	1.957
18	28	0	50	1.002
23	36	0.166	65	1.298
r	0.919		0.944	
r (without 12)	0.743		0.977	

Table 5. Correlation coefficient r between x = number of permitted tables for $1/100 \le s_m/N \le 1/k$ and y = percent of accessible data values. Database 11' is omitted because of its strong dependency on Database 11. Database 13 is omitted because of its steep risk line, implying low risks for $s_m/N \le 1/10$.

are computed both with and without Database 12.) Note that although the correlations are quite high, the relationship between the number of tables and accessible data values is unlikely to remain linear for arbitrarily large numbers of tables or for databases with steep risk lines. Nevertheless, the data suggests that if we want to use larger values of k with s_m / N (augmenting the control with, say, perturbation techniques), the number of permitted tables could be used as a criterion for selecting k.

We conclude that s_m / N can control disclosure. If we want to minimize information loss, we can pick larger values for k, and augment the control with some other protection technique. Aside from the order control, it is the simplest of the controls to implement.

Minimum frequency ($\Pi rmin$). The $\Pi rmin$-criterion is highly secure but overly restrictive. If we relax the criterion by decreasing k, its ability to approximate the $m+1$-rule is likely to deteriorate since it uses only the smallest relative frequency of each attribute. Moreover, given that we are going to use frequency distributions at all, we can do much better with explicit risk estimation, which uses the complete distributions.

Explicit risk estimation. Table 2 shows that explicit risk estimation can effectively control disclosure without falsely restricting too many tables. Note that version 2 of the control, which is based on the risk of a table (rather than its parents), is similar to $\Pi rmin$ with $k = 1$. Although both are overly restrictive, with explicit risk estimation we could relax the restriction criterion and apply other security measures.

Explicit risk estimation has the advantage over s_m / N of treating the databases more uniformly; that is, the same parameter z causes similar risk and information loss in all databases. By taking into account higher dimensional frequency distributions, even closer estimates of the risk are possible [Schl82a].

It has the disadvantage over s_m / N of requiring more information about the databases, namely their frequency distributions. Moreover, if the decision to restrict a table is based on the estimated risks of the parents (version 1 of the control), this requires more computation than when the decision is based on the estimated risk of the table itself (version 2). Although the control is clearly more costly that s_m / N, the cost need not be prohibitive; if only 1-dimensional frequency distributions are used, for example, these can be computed and stored periodically as needed to account for the dynamics of the database.

The next two sections describe enhancements and extensions to table restriction controls. Section 6 discusses possible security leaks, how these leaks might be plugged, and security techniques that would allow the restriction criterion to be relaxed. Section 7 considers extensions to the criterion for handling SUMs and other higher-order statistics (i.e., not just COUNTs).

6. Plugging the Leaks

The disclosure risks of the preceding section were calculated under the assumption that the $m+1$-rule prevents disclosure. In fact, it does not prevent all types of disclosure. Figure 9 shows how it can be circumvented; the example is adapted from one by Olsson [Olss75a]. At the Swedish Bureau of Statistics this has been nicknamed the "problem of the magical zeros" because of the strategic location of the zeros in the 3-table. Since the 1-tables for this data have no identifications, the 2-tables are permitted by the $m+1$-rule. The 3-table, on the other hand, is restricted since a female veterinarian is identified in the 2-table over sex and occupation. Nevertheless, all values in the 3-table can be deduced from the permitted 2-tables. (In the original example, the 2-tables have no identifications and, therefore, look innocent. Yet it is still possible to

Problem of the magical zeros: Given the three 2-tables below over all pairs of the attributes sex, occupation, and tax honesty, the 3-table over all attributes can be deduced. Abbreviations: phy = physician, den = dentist, vet = veterinarian, f = female, m = male.

	f	m
tax dodger	5	28
honest taxpayer	14	23

	phy	den	vet
tax dodger	10	19	4
honest taxpayer	24	6	7

	phy	den	vet
f	7	11	1
m	27	14	10

Solution: Since there are 19 dentists that dodge taxes but only 5 female tax dodger, all 14 male dentists must dodge taxes. From this, the remaining entries can be deduced, giving:

	tax dodgers			honest tax payers		
	phy	den	vet	phy	den	vet
f	0	5	0	7	6	1
m	10	14	4	17	0	6

Figure 9. Deducing a 3-table from 2-tables (adapted from Olsson [Olss75a]).

deduce the 3-table owing to the magical zeros.)

To prove that a restricted $m+1$-table is secure from exact disclosure, we must prove it is **m-transformable**; that is, there is at least one other $m+1$-table derivable from the permitted m-tables [Dale79a, Schl81a]. Unfortunately, proving that a table is m-transformable is an NP-complete problem [Reis77a]; fortunately, the complementary problem of compromising an $m+1$-table by magical zeros is also difficult for large tables.

Even if we prove m-transformability, the data may still be vulnerable to **negative** and **approximate** disclosure [Dale77a, Fell72a, Fell74a, Haq75a, Olss75a, Rapa75a, U.S.78a]. Similar deductions are possible using "pseudo trackers" [Schl75a]. Dynamic databases are also vulnerable to update attacks [Chin79a, Chin81a, Chin82a, Ozso81a, Yu77a].

We now mention three methods of strengthening table restriction techniques to reduce the risk of the above disclosures as well as those caused by releasing tables that violate the $m+1$-rule (i.e., the false permissions in Table 3). The first method is to couple the technique with a simple **output perturbation** technique. Likely candidates are **systematic** or **random rounding** [Achu79a, Alag83 , Fell74a, Narg72a, Schl81b], **random sample queries** [Denn80b], and **random data perturbation** [Beck80a]. (**Controlled rounding** [Caus79a, Cox81a, Dale81a] is presently expensive for query processing systems.) The amount of noise introduced could increase with s_m / N or the explicit risk estimator T. This would allow the release of higher-order statistics than could be permitted under table restriction alone.

Many perturbation techniques are insufficient as stand-alone controls. Systematic rounding, for example, can be circumvented by individual trackers [Schl77a, Schl81b]; random perturbation schemes are frequently vulnerable to error removal by averaging [Denn80b, Schw77a]. A control combining perturbation with table restriction should provide an acceptable level of security for most applications.

The second method of strengthening table restriction is **grouping** or **rolling up** [Fell72a, Fell74a, Olss75a]. Figure 10 shows how this strategy could be used to protect the set with cardinality 1 in Figure 4 of Section 4. Here, the 2-sets for attribute values b_1 and b_2 are merged. If groups are defined by ranges of values, then the individual table cells correspond to range queries for fixed ranges. Grouping has the effect of decreasing the estimated disclosure risk (e.g., s_m / N or $\hat{T}$), thereby permitting tables that otherwise would be restricted. This technique is often rejected for off-line systems because it hinders comparability of different tables [Fell72a]. In on-line systems, users can choose from alternative groupings, though arbitrary groupings cannot be allowed [Schl76a].

		B	
		b_1+b_2	b_3
	a_1	4	6
A	a_2	11	5
	a_3	8	6

Figure 10. Grouping or rolling up applied to Figure 4.

The third method of strengthening table restriction is to adjust the threshold parameter for restriction (i.e., $1/k$ for s_m / N; z for explicit risk estimation) to the attributes of the table. In particular, tables over sensitive attributes could be given lower thresholds than those with less sensitive attributes. Similarly, tables over attributes in the "identification subbase"

[Schl80a] could be given lower thresholds since these attributes are likely to be used for record identification.

Threat monitoring is a valuable tool with any control for determining whether a user has attempted to obtain restricted data [Hoff70a]. With the s_m / N-criterion, for example, it might be useful to record for each restricted query the value s_m / N of the query and the attributes of the characteristic formula.

7. General Additive Statistics

We now turn to general d-order additive statistics of the form $q(C) = f(C,D)$, where f is a statistical function (e.g., SUM), C is an m-set over m characteristic attributes, and D is a set of data attributes, $d-m$ of which do not appear in C (see Section 2). Note that whereas such statistics are elements of an m-table over the characteristic attributes, they can be derived from the d-table of cardinalities over C and D. This suggests two alternatives for applying table restriction to $f(C,D)$:

1. Apply it to all d attributes in $C \cup D$, or

2. Apply it only to the m attributes in C.

Approach (1) can be unacceptably restrictive. Consider, for example, the query

$$q(C) = \text{SUM}((\text{SEX}=\text{MALE})\ \&\ (\text{MAJOR}=\text{EE}),\ \text{GRADEPOINT}))\ .$$

We have |SEX| = 2. Suppose that MAJOR has 20 values and that the domain of grade-points is {0.00, 0.01, ..., 4.00}. Then

$$s_3 = |\text{SEX}| * |\text{MAJOR}| * |\text{GRADEPOINT}| = 2 * 20 * 401 = 16{,}040.$$

Now, suppose the relative table size control is used with $k = 10$; that is, the upper bound for s_m / N is $1/10$. Then $q(C)$ will be restricted (or perturbed) unless N is at least 160,400 (an unlikely possibility in any university!). But as long as no student is uniquely identified by SEX and MAJOR, the grade-points in the 2-table defined by SEX and MAJOR will not disclose any particular student's grade-point. Approach (2) seeks to avoid such unnecessary restrictions by using only the characteristic attributes in the s_m / N-criterion; in this case, $s_2 =$ |SEX| * |MAJOR| = 40. This would permit the 2-table of grade-points for $N \geq 400$.

Approach (2), on the other hand, can be overly permissive. If there is a single male student majoring in EE, then the above statistic from the 2-table over SEX and MAJOR discloses his exact grade-point. This problem does not arise with statistics for COUNTs because the cardinalities are obtained from the 3-table over SEX, MAJOR, and GRADEPOINT, which would be rejected by the s_m / N-criterion.

Releasing additive statistics computed over m-sets with cardinalities greater than 1 can also lead to disclosure. For example, consider a query SUM(C, A), where $|C| = 2$. One of the two individuals represented in the statistic can deduce the value of the other (assuming his identity is known) by subtracting his own value for A from the statistic. Economic data is particularly vulnerable to such compromise because the identities of the respondents is frequently known [Fell72a, Olss75a, U.S.78a ,Cox80a].

For some statistical functions, an intruder can deduce the values used to compute $q(C)$ even when he is not represented in C. Suppose, for example, that $|C|$ is known. Define

$$q_e(C, A) = \sum_{i \in C} x_i^e , \qquad \text{for } e = 1,\ldots,|C| , \tag{7.1}$$

where x_i is the value of attribute A in record i. From the $|C|$ statistics (7.1), the values x_i for the records i in C can be computed [Dale82a]. Assuming statistical functions with exponents up to 4 are available, this strategy is effective for compromising sets with cardinalities of 4 (or less).

A similar situation holds for additive statistics of the form

$$q(C, A, B) = \sum_{i \in C} x_i y_i \; . \tag{7.2}$$

Assume $|C| = 2$ and that the two records are $R_1 = (x_1, y_1, z_1)$ and $R_2 = (x_2, y_2, z_2)$. Then R_1 and R_2 can be deduced from the set of statistics

$$\{\sum x_i, \sum y_i, \sum z_i, \sum x_i y_i, \sum x_i z_i, \sum y_i z_i \mid i \in C\}.$$

Note that such deductions become more difficult if instead of the sums (7.1) or (7.2), nonadditive statistics such as variance and covariance are released.

We need a criterion that is less restrictive than approach (1), but more restrictive than (2). The above shows that such a criterion should depend on the statistical function as well as the attributes of a query. We believe that the best strategy is to follow approach (2), but with tighter thresholds for complex statistics to avoid 1-cells and possibly 2-cells.

8. Summary

We have studied memoryless restriction techniques operating at the cell level and table level in the lattice. We began our investigation of cell-level techniques by noting that a query-set-size control, while valuable, is easily subverted by trackers. We then investigated the feasibility of an implied queries control, which restricts a statistic over attributes $A_1, \ldots, A_m$ if it could be used to compute a sensitive statistic over $A_1, \ldots, A_m$. We saw that this requires inspecting the sizes of at least 2^m elementary m-sets in the m-table over $A_1, \ldots, A_m$. Moreover, unless we restrict the syntax of characteristic formulas, we must inspect all elementary m-sets in the table, and if any one of these sets is sensitive, then the entire table of statistics must be restricted.

Inspecting all m-sets in a table with s_m values can be costly for large s_m.

This led us to look for more efficient table restriction techniques. We considered four candidates: order, relative table size (s_m / N), minimum frequency ($\Pi rmin$), and explicit risk estimation. Of these candidates, relative table size and explicit risk estimation look the most promising. Either technique could be combined with a simple output perturbation scheme for increased security.

Acknowledgements

We are indebted to L. Zick for Figures 6-8 and the programs used in our empirical studies, and to W. Gaus for his help in getting the time and resources to undertake this study. We also thank A. Shoshani and the referees for their constructive comments.

References

Achu79a. Achugbue, J. O. and Chin, F. Y., "The Effectiveness of Output Modification by Rounding for Protection of Statistical Databases," *INFOR* **17**(3), pp. 209-218 (Mar. 1979).

Alag83. Alagar, V., "Range Response: An Output Modification Technique and an Analysis of its Effectiveness for Security of Statistical Databases," Dept. of Computer Science, Concordia Univ., Montreal, Canada (D).

Beck80a. Beck, L. L., "A Security Mechanism for Statistical Databases," *ACM Trans. on Database Syst.* **5**(3), pp. 316-338 (Sept. 1980).

Bloc76a. Block, H. and Olsson, L., "Bakvagsidentifiering," *Statistisk Tidskrift* **14**, pp. 135-144 (1976).

Caus79a. Causey, B., "Approaches to Statistical Disclosure," in *Proc. Amer. Stat. Assoc., Soc. stat. Sec.*, Washington, D. C. (1979), 306-309.

Chin79a. Chin, F. Y. and Ozsoyoglu, G., "Security in Partitioned Dynamic Statistical Databases," pp. 594-601 in *Proc. IEEE COMPSAC Conf.* (1979).

Chin81a. Chin, F. Y. and Ozsoyoglu, G., "Statistical Database Design," *ACM Trans. on Database Syst.* **6**(1), pp. 113-139 (Mar. 1981).

Chin82a. Chin, F. Y. and Ozsoyoglu, G., "Auditing and Inference Control in Statistical Databases," *IEEE Trans. Software Eng.* **SE-8**(6), pp. 574-582 (Nov. 1982).

Cox78a. Cox, L. H., "Suppression Methodology and Statistical Disclosure Control," Confidentiality in Surveys, Report No. 26, Dept. of Statistics, Univ. of Stockholm, Stockholm, Sweden (Jan. 1978).

Cox80a. Cox, L. H., "Suppression Methodology and Statistical Disclosure Control," *J. Amer. Stat. Assoc.* **75**(370), pp. 377-385 (June 1980).

Cox81a. Cox, L. H. and Ernst, L. R., "Controlled Rounding," U.S. Bureau of the Census, Washington, D.C. (Jan. 1981).

Dale77a. Dalenius, T., "Towards a Methodology for Statistical Disclosure Control," *Statistisk tidskrift* **15**, pp. 429-444 (1977).

Dale79a. Dalenius, T. and Reiss, S. P., "Data-Swapping -- A Technique for Disclosure Control," pp. 191-196 in *Proc. of the Section on Survey Research Methods*, Amer. Stat. Assoc., Washington, D.C. (1979).

Dale81a. Dalenius, T., "A Simple Procedure for Controlled Rounding," *Statistisk Tidskrift*(3), pp. 202-208 (1981).

Dale82a. Dalenius, T. and Denning, D., "A Hybrid Scheme for Release of Statistics," *Statistisk Tidskrift*(2), pp. 97-102 (1982).

DeJo83. DeJonge, W., "Compromising Statistical Databases Responding Only to Queries About Means.," *ACM Trans. on Database Systems* 8(1), pp.60-80 (Mar. 1983).

Denn79a. Denning, D. E., Denning, P. J., and Schwartz, M. D., "The Tracker: A Threat to Statistical Database Security," *ACM Trans. on Database Syst.* **4**(1), pp. 76-96 (Mar. 1979).

Denn80a. Denning, D. E. and Schlörer, J., "A Fast Procedure for Finding a Tracker in a Statistical Database," *ACM Trans. on Database Syst.* **5**(1), pp. 88-102 (Mar. 1980).

Denn80b. Denning, D. E., "Secure Statistical Databases Under Random Sample Queries," *ACM Trans. on Database Syst.* **5**(3), pp. 291-315 (Sept. 1980).

Denn81a. Denning, D. E., "Restricting Queries That Might Lead to Compromise," *Proc. IEEE 1981 Symp. on Security and Privacy*, pp. 33-40 (Apr. 1981).

Denn82a. Denning, D. E., *Cryptography and Data Security*, Addison-Wesley, Reading, Mass. (1982).

Denn83a. Denning, D. E., "A Security Model for Statistical Databases," *Proc. 2nd Intl. Workshop on Statistical Database Management*, pp. 368-390 (Sept. 1983).

Dobk79a. Dobkin, D., Jones, A. K., and Lipton, R. J., "Secure Databases: Protection Against User Inference," *ACM Trans. on Database Syst.* **4**(1), pp. 97-106 (Mar. 1979).

Fell72a. Fellegi, I. P., "On the Question of Statistical Confidentiality," *J. Amer. Stat. Assoc.* **67**(337), pp. 7-18 (Mar. 1972).

Fell74a. Fellegi, I. P. and Phillips, J. L., "Statistical Confidentiality: Some Theory and Applications to Data Dissemination," *Annals Econ. Soc'l Measurement* **3**(2), pp. 399-409 (April 1974).

Frie80a. Friedman, A. D. and Hoffman, L. J., "Towards a Fail-Safe Approach to Secure Databases," pp. 18-21 in *Proc. 1980 Symp. on Security and Privacy*, IEEE Computer Society (Apr. 1980).

Haq74a. Haq, M. I., "Security in a Statistical Data Base," *Proc. Amer. Soc. Info. Sci.* **11**, pp. 33-39 (1974).

Haq75a. Haq, M. I., "Insuring Individual's Privacy from Statistical Data Base Users," pp. 941-946 in *Proc. NCC,*, Vol. 44, AFIPS Press, Montvale, N.J. (1975).

Hoff70a. Hoffman, L. J. and Miller, W. F., "Getting a Personal Dossier from a Statistical Data Bank," *Datamation* **16**(5), pp. 74-75 (May 1970).

Kam77a. Kam, J. B. and Ullman, J. D., "A Model of Statistical Databases and their Security," *ACM Trans. on Database Syst.* **2**(1), pp. 1-10 (Mar. 1977).

Narg72a. Nargundkar, M. S. and Saveland, W., "Random Rounding to Prevent Statistical Disclosure," *Proc. Amer. Stat. Assoc., Soc. Stat. Sec.*, pp. 382-385 (1972).

Olss75a. Olsson, L., "Protection of Output and Stored Data in Statistical Databases," ADB-Information, 4, Statistika Centralbyran, Stockholm, Sweden (1975).

Ozso81a. Ozsoyoglu, G. and Ozsoyoglu, M., "Update Handling Techniques in Statistical Databases," in *Proc. First LBL Workshop on Statistical Database Management*, Lawrence Berkeley Lab, Berkeley, CA (Dec. 1981), 249-284.

Rapa75a. Rapaport, E. and Sundgren, B., "Output Protection in Statistical Databases," S/SYS-E04, Nat. Central Bur. Stat., Stockholm, Sweden (1975). (Invited paper, Warsaw Meeting Int. Stat. Inst., Oct. 1975)

Reis77a. Reiss, S. P., "Statistical Database Confidentiality," Confidentiality in Surveys, Rept. No. 25, Dept. of Stockholm, Univ. of Stockholm, Stockholm (Nov. 1977).

Reis78a. Reiss, S. P., "Medians and Database Security," pp. 57-92 in *Foundations of Secure Computation*, ed. R. A. DeMillo et al., Academic Press, New York (1978).

Sand77a. Sande, G., "Towards Automated Disclosure Analysis for Establishment Based Statistics," Statistics Canada (1977).

Schl75a. Schlörer, J., "Identification and Retrieval of Personal Records from a Statistical Data Bank," *Methods Inf. Med.* **14**(1), pp. 7-13 (Jan. 1975).

Schl76a. Schlörer, J., "Confidentiality of Statistical Records: A Threat Monitoring Scheme for On Line Dialogue," *Methods Inf. Med.* **15**(1), pp. 36-42 (1976).

Schl77a. Schlörer, J., "Confidentiality and Security in Statistical Data Banks," pp. 101-123 in *Data Documentation: Some Principles and Applications in Science and Industry; Proc. Workshop on Data Documentation*, ed. W. Guas and R. Henzler, Verlag Dokumentation, Munich, Germany (1977).

Schl80a. Schlörer, J., "Disclosure from Statistical Databases: Quantitative Aspects of Trackers," *ACM Trans. on Database Syst.* **5**(4), pp. 467-492 (Dec. 1980).

Schl81a. Schlörer, J., "Security of Statistical Databases: Multidimensional Transformation," *ACM Trans. on Database Syst.* **6**(1), pp. 95-112 (Mar. 1981).

Schl81b. Schlörer, J., "Security of Statistical Databases: Ranges and Trackers," Klinische Dokumentation, Universitat Ulm, Ulm, W. Germany (Nov. 1981).

Schl82a. Schlörer, J. and Zick, L., "Empirical Investigations on the Identification Risk in Statistical Databases," Klinische Dokumentation, Universitat Ulm, Ulm, W. Germany (June 1982).

Schl83. Schlörer, J., "Information Loss in Partitioned Statistical Databases," *Computer J.* **26**(3), pp.218-223 (1983).

Schl84. Schlörer, J., "Insecurity of Set Controls for Statistical Databases," *Information Processing Letters* **18**(2), pp.67-71 (Feb. 1984).

Schw77a. Schwartz, M. D., "Inference from Statistical Data Bases," Ph.D. Thesis, Computer Sciences Dept., Purdue Univ., W. Lafayette, Ind. (Aug. 1977).

Selb74a. Selbmann, H. K., "Bitstring Processing for Statistical Evaluation of Large Volumes of Medical Data," *Methods Inf. Med.* **13**(1), pp. 61-64 (Jan. 1974).

Sund73a. Sundgren, B., "An Infological Approach to Data Bases," Urval Nr. 7, Nat. Central Bur. Stat. and Univ. Stockholm, Stockholm, Sweden (1973).

Sund78a. Sundgren, B., "RAM - A Framework for a Statistical Production System," S/SYS - E02, Nat. Central Bur. Stat., Stockholm, Sweden (1978).

U.S.78a. U.S.,, "Report on Statistical Disclosure and Disclosure-Avoidance Techniques," U.S. Dept. of Commerce, U.S. Government Printing Office, Washington, D.C. (1978).

Wehr83. Wehrle, E. and Schlörer, J., "The Partner Algorithm for Protecting Statistical Databases - Extended Abstract," pp. 134-145 in *GI - 13. Jahrestagung, Proceedings*, ed. I. Kupka, Springer, Berlin (1983).

Wehr84. Wehrle, E., "Der Partner-Algorithmus zum Schutz der Vertraulichkeit Persönlicher Angaben bei Statistischen Auswertungen," Doc. diss., Universitat Ulm, Ulm, W. Germany (D).

Yao79a. Yao, A. C., "A Note on a Conjecture of Kam and Ullman Concerning Statistical Databases," *Info. Proc. Let.* **9**(1), pp. 48-50 (July 1979).

Yu77a. Yu, C. T. and Chin, F. Y., "A Study on the Protection of Statistical Databases," *Proc. ACM SIGMOD Int. Conf. Management of Data*, pp. 169-181 (1977).