

New Approaches to Disclosure Limitation While Answering Queries to a Database: Protecting Numerical Confidential Data Against Insider Threat Based on Data or Algorithms

Robert Garfinkel¹, Ram Gopal¹, Daniel Rice²

¹ School of Business Administration, University of Connecticut, Storrs, Connecticut, U.S.A.

² School of Business and Management, Loyola College in Maryland, Baltimore, Maryland, U.S.A.

E-mail: robert.garfinkel@business.uconn.edu; ram.gopal@business.uconn.edu; DRice2@loyola.edu

Abstract

A practical method for giving unlimited, correct, numerical responses to ad-hoc queries to an on-line database, while not compromising confidential numerical data, has been developed by (Gopal et al. 2002) and is called *Confidentiality via Camouflage* (CVC). Responses are in the form of intervals that are guaranteed to contain the exact answer. Virtually any imaginable query type can be answered and although sharing of query answers among users presents no problem, the threat of insider information is real. In this work we identify two distinct types of insider information, depending on whether the knowledge is of data in the confidential field or of the algorithmic process that is used to answer queries. We show that different realizations of CVC can protect against one type of insider threat or the other, while a combination of realizations can be used if the database administrator is not able to specify the type of threat that is present. Various strategies for dealing with cases where a user poses both types of threats are also presented. Computational experience relates the degradation of answer intervals that can be expected based on the type of threat that is protected against and indicates that, in general, algorithmic threat causes the greatest degradation.

1 Introduction

Historically there have been three approaches to protection of a confidential vector of numerical data in a database from disclosure through answers to user queries. They are first differentiated by whether or not the user is afforded deterministically correct answers to queries that are functions of that field. If not, a simple and effective approach is through *perturbation* of the data, typically so as to retain desired statistical properties. Examples of the perturbation approach are (Duncan and Mukherjee 2000, Muralidhar et al. 1995, and Traub et al. 1984). If the user does require deterministically correct answers, the next level of differentiation is whether these answers are required to be exact, i.e. a number, or whether an interval answer will suffice. The first requirement leads naturally to a class of techniques known as *query restriction*. That is, since exact answers to queries provide the user with such powerful information, it may become necessary to refuse to answer certain queries at some stage in the process to avoid disclosure of a confidential datum. These approaches are typified by (Chin and Ozsoyoglu 1982 and Gopal et al. 1998). If deterministically correct, and hopefully small, interval answers suffice, a model known as *Confidentiality via Camouflage* (CVC) has been presented in (Gopal et al. 2002) that allows for unlimited answers to any conceivable query types. In this work we consider extensions to the realizations of CVC given in (Gopal et al. 2002) that focus on the types of (insider) knowledge that the users may have in addition to the answers to queries.

In particular this work distinguishes between the knowledge of confidential data or of the process used to answer queries. That distinction is an essential departure from previous work in the area of security of a confidential field in a database. The threat of insider knowledge of data is often treated in an ad-hoc method in the literature, although (Garfinkel et al. 2002) define it precisely and show how to protect against it when the confidential data is categorical. To our knowledge the threat of knowledge of the algorithmic process of answering queries has not been addressed before.

The definition of protection can also vary among different models. As in (Gopal et al. 2002) we consider a form of deterministic interval protection in which there is associated with the confidential n -vector $\mathbf{a} := (a_1, \dots, a_n)$, two other n -vectors $\mathbf{l} := (\ell_1, \dots, \ell_n)$ and $\mathbf{u} := (u_1, \dots, u_n)$, such that $\mathbf{l} \leq \mathbf{a} \leq \mathbf{u}$. If a user is able to determine that $a_i \in \Omega_i \subseteq R$, then record i is protected if and only if

$$\Omega_i \cap (-\infty, \ell_i] \neq \phi \text{ and } \Omega_i \cap [u_i, +\infty) \neq \phi. \quad (1)$$

That is, no user can determine that $a_i > \ell_i$ or $a_i < u_i$ for any i .

We illustrate all concepts on Table 1 from (Gopal et al. 2002). Note that the table contains entries in the three non confidential fields “Job”, “Age”, and “Company”, as well as in the confidential field “Salary”. We assume that the nonconfidential fields can be used as a basis for answering queries, i.e. “the maximum salary of employees of Company A”. A good deal of attention was devoted to these fields in (Gopal et al. 2002) where it was shown that there are advantages, in terms of the size of the resulting answer intervals, to renumbering the records based on them. In this work we do not pay special attention to these fields and simply assume that there are enough of them to enable a user to essentially choose any set T as the “target set” of a query, e.g. $T = \{1, 3, 5, 6, 9, 11, 13, 14\}$ can be asked with “Company A or Manager”. In terms of protection, note that for instance, from Table 1, a_1 is protected on both sides by an interval of range seven that is asymmetric around it. On the other hand a_2 is protected only from below while a_{14} requires no protection.

Record	Name	Job	Age	Company	a = Salary	l	u
1	Robinson	Manager	27	A	55	53	60
2	Reese	Trainee	42	B	31	29	31
3	Furillo	Manager	63	C	107	99	110
4	Campanella	Trainee	28	B	28	28	31
5	Cox	Manager	55	B	63	53	64
6	Snider	Manager	57	A	82	78	82
7	Koufax	Trainee	21	D	29	28	30
8	Newcombe	Trainee	32	C	31	29	31
9	Hodges	Manager	35	B	60	60	63
10	Branca	Trainee	36	D	27	26	27
11	Loes	Manager	47	B	47	46	50
12	Roe	Trainee	28	D	32	31	34
13	Reiser	Manager	64	A	94	91	100
14	Gilliam	Manager	46	C	51	51	51

Table 1

It is assumed that the parameters (ℓ_i, u_i) are provided by the DBA, possibly based on at least partial input from the subjects of the database. For instance the DBA may guarantee that no user will be able to determine a_i within a given percentage of its value. There are many (ℓ_i, u_i) pairs that satisfy that condition and the DBA will choose one. He will be motivated to use some degree of randomization and, in particular, will not want to have $[\ell_i, u_i]$ routinely be a symmetric interval around a_i . The latter would be particularly dangerous if $\Omega_i = [\ell_i, u_i]$, which is likely to occur in some realizations of CVC. Then if the user estimated a_i as the midpoint of Ω_i , the estimate would be exact.

Section 2 describes CVC and defines the two basic types of insider threat. Two different realizations of CVC are the subjects of Sections 3 and 4. The first is that of (Gopal et al. 2002) and is shown to be safe against insider knowledge of the algorithmic process but not of confidential data. The reverse is true of the second realization, introduced in this work. Section 5 indicates how the two realizations of CVC can be combined in various ways and indicates the benefits achieved from this combination in terms of protection or answer quality or both. Computational results are presented in Section 6 and conclusions and future research in the ultimate section.

2 An Overview of CVC and Analysis of Threats

2.1 Overview of CVC

In CVC the confidential vector n -vector \mathbf{a} is “camouflaged” by making it part of the relative interior of a compact set Π of n -vectors. Then every query $q := f(\mathbf{a})$ is answered with the interval $[q^-, q^+]$, such that

$$q^- \leq f^- \text{ and } q^+ \geq f^+ \tag{2}$$

where

$$f^- := \min\{f(\mathbf{x}) : \mathbf{x} \in \Pi\} \tag{3}$$

and

$$f^+ := \max\{f(\mathbf{x}) : \mathbf{x} \in \Pi\}. \quad (4)$$

Let $x_i^- := \min\{x_i : \mathbf{x} \in \Pi\}$ and $x_i^+ := \max\{x_i : \mathbf{x} \in \Pi\}$. Then the user will not be able to determine a_i , *solely from the answers to queries*, more precisely than $a_i \in [x_i^-, x_i^+]$. It follows that as long as

$$\exists (\mathbf{x}, \mathbf{y}) \in \Pi \ni x_i \leq \ell_i \text{ and } y_i \geq u_i \text{ for all } i \quad (5)$$

every subject is protected. There are two main classes of issues that remain with respect to implementation of CVC, based on whether or not it is assumed that the user gains all of her knowledge from answers to queries or if she has access to some other information.

If the user knows more than just the answers to queries, protection of the confidential data becomes much more complex and is dependent on what kinds of “insider” information the user has. These issues are addressed in Section 2.3. If not, the first remaining issue is the structure of the set Π . We assume only that Π is compact and obeys (5).

Once the structure of Π is chosen there remains the question of how to solve (3), (4). Of course these depend on the function f . In (Gopal et al. 2002) only minimal access, low order polynomial algorithms for (3), (4) are considered because the system is assumed to be running on line. We make the same assumption here. Clearly the form of the set Π will influence the ease of determination of f^- and f^+ and the choice of exact algorithm or heuristic to solve (3), (4) will influence the quality of the answers given. In general a smaller interval is preferred by the user to a larger interval. Let e be the exact, correct answer to a query. To measure the quality of an answer based on (3), (4) we use the measure $h := (q^+ - q^-)/e$, so that quality is inversely related to h .

2.2 Threat Analysis for CVC

As indicated in Section 2.1, CVC provides the desired protection (1) as long as the user does not have access to additional information, i.e. if all the user knows is that for every query q that has been answered, $f(\mathbf{a}) \in [q^-, q^+]$. There are basically two types of additional information that a user could have. We will refer to them as *insider data information* and *insider algorithm information*. At the minimum insider data information could be in the form

of better answers to queries than those provided by CVC, i.e. the user knows, for some query function f , that $f(\mathbf{a}) > q^-$ or $f(\mathbf{a}) < q^+$. The extreme case is that the user knows one or more elements of the confidential vector.

On the other hand insider algorithm information can be further classified as knowledge of either the process of the algorithm or its parameters. For CVC we can define five different levels of such information, namely:

Insider Algorithm Process Information

- a. The general model (3), (4);
- b. In addition to (a), whether a given query type is answered exactly or heuristically;
- c. In addition to (a) and (b), the type of heuristic used if appropriate;
- d. In addition to (a), the general structure of $\mathbf{\Pi}$.

Insider Algorithm Parameter Information

- e. In addition to (d), any parameters that determine $\mathbf{\Pi}$ and the relationship of \mathbf{a} to $\mathbf{\Pi}$.

In the remainder of the paper we focus on the various realizations of possible threats, and measures to safeguard against them that are dependent on the particular structure of $\mathbf{\Pi}$ that is chosen. We will see that it is possible to focus the choice of $\mathbf{\Pi}$ on whether insider data information or insider algorithm information is seen to be the main threat or even if the DBA has no way to decide which type or types of information to guard against.

3 CVC-POL: Polytope

In (Gopal et al. 2002) the compact set Π is restricted to be a polytope so that we will now refer to that realization of CVC as CVC-POL. In particular the polytope is the convex hull of k extreme n -vectors $\mathbf{P} = \{\mathbf{P}^1, \dots, \mathbf{P}^k\}$, where $k \geq 3$ and typically $k \leq 6$. Two different classes of polytopes are discussed. One is especially appropriate for SUM queries in that it is designed to be able to answer a subclass of these queries exactly. The other is more robust in that it tends to give better answers, as defined by tighter intervals, to most standard classes of queries. We restrict our attention to the second class in this work.

In particular $k = 3$ and \mathbf{P} is constructed as follows. The elements of \mathbf{l} and \mathbf{u} are systematically (see (Gopal et al. 2002) for details) assigned to $\mathbf{P}^1 :=$

(p_1^1, \dots, p_n^1) and $\mathbf{P}^2 := (p_1^2, \dots, p_n^2)$ so that for all i , $p_i^1 = \ell_i$ and $p_i^2 = u_i$ or vice versa. Then λ_1, λ_2 are chosen such that $\lambda_1, \lambda_2 > 0$ and $\lambda_1 + \lambda_2 < 1$. Finally to make \mathbf{a} part of the relative interior of Π , $\mathbf{P}^3 = (\mathbf{a} - \lambda_1 \mathbf{P}^1 - \lambda_2 \mathbf{P}^2) / (1 - \lambda_1 - \lambda_2)$. For example, arbitrarily choosing $\lambda_1 = .2$ and $\lambda_2 = .3$, \mathbf{P} is given in Table 2.

Record	\mathbf{P}^1	\mathbf{P}^2	\mathbf{P}^3	\mathbf{a}
1	60	53	54.2	55
2	31	29	32.2	31
3	99	110	108.4	107
4	28	31	26.2	28
5	53	64	66.4	63
6	78	82	83.6	82
7	30	28	29.2	29
8	29	31	31.8	31
9	63	60	58.8	60
10	26	27	27.4	27
11	46	50	45.6	47
12	34	31	31.8	32
13	91	100	91.6	94
14	51	51	51	51

Table 2

Since this definition of Π satisfies (5) it gives at least the required protection (1) but additional protection is often provided from the vector \mathbf{P}^3 , as is the case for the records 2,4,5,6,8,9,10,11 in Table 2.

Four classes of algorithms for solution of (3), (4) over the two variables λ_1, λ_2 are considered in (Gopal et al. 2002). All algorithms are restricted to be *minimal access*, i.e. each record has to be accessed no more frequently than would be the case if the query were answered exactly since, as a general rule, a database access requires orders of magnitude more time than a calculation performed in the main memory. The solution types are:

Extreme Point Solution: For queries having f either convex or concave (e.g. SUM, MIN, VARIANCE), (3) or (4) or both are solved by extreme points Π which are the elements of \mathbf{P} . For example, suppose the query is “The mean salary of the employees of Company B”, so that $T = \{2, 4, 5, 9, 11\}$. Then from Table 2, $e = 45.8$, $I = [44.2, 46.8]$, $h = .0568$.

Very Efficient Algorithms: In some cases where (3) or (4) is not solved by an extreme point of Π (e.g.(3) for VARIANCE), exact solution can be had using a minimal access algorithm with time bounded by a low order polynomial in k . This is very efficient especially since $k = 3$.

Grid search: If f is a continuous function, which is true of many standard statistical queries, a fine grid search over $\lambda_1 \in [0, 1]$ and $\lambda_2 \in [0, 1 - \lambda_1]$ will invariably provide very good solutions to (3), (4). This option is used, for instance, for the R^2 statistic for simple linear REGRESSION queries.

Bounding Heuristics: A final option for all remaining queries is to design fast, minimal access heuristics which yield $[q^-, q^+]$ as given in (2). For instance these are developed in (Gopal et al. 2002) for PERCENTILE and COUNT queries.

For each of a number of standard query types examples of one or the other of the above algorithms are given in (Gopal et al. 2002), and generally provide very good answers for a variety of simulated databases. These algorithms will not be described here, but their answers will be compared to those of another realization of CVC in Section 4.

3.1 Threat analysis for CVC-POL

It is easy to see that CVC-POL is vulnerable to insider data information. For instance it is possible that CVC-POL gives exact answers to SUM queries, so that a user who knows $t - 1$ elements of \mathbf{a} determines the remaining element from the answer to the query. For an example from Table 2 where the answer is not exact but confidentiality is nevertheless violated, suppose a user knows that $a_7 = 29$ and asks SUM with $T = \{7, 8\}$. The answer would be $[59, 61]$ from which the user establishes that $a_8 \geq 30$, which violates the lower bound constraint for Subject 8. On the other hand CVC-POL is only marginally vulnerable to insider algorithm information. There is no danger of loss of confidentiality from (a)-(c) of Section 2.2. Knowledge of the general structure of Π is also safe. That is, the user can know that Π is a polytope and that \mathbf{a} is interior to Π . Finally, the parameters of CVC-POL are k and $(\lambda_1, \dots, \lambda_k)$. The user can safely know k and even the k extreme points themselves and it is easy to see that this knowledge can be achieved, without any insider data information, through a series of linear queries.

If, however, the user know (a)-(d) and $\mathbf{P}^1, \dots, \mathbf{P}^k$, knowledge of the remaining parameters $(\lambda_1, \dots, \lambda_k)$ would allow for determination of \mathbf{a} in its entirety.

On the other hand we note that knowledge of $(\lambda_1, \dots, \lambda_k)$ can be denied to all humans, including the DBA, by simply having the computer program randomly choose this k -vector. Thus, while knowledge of these parameters may not be considered a real danger, insider data information consisting of knowledge of any k elements of \mathbf{a} , along with (a)-(d) would permit a user to obtain $(\lambda_1, \dots, \lambda_k)$ and thus the exact values of the remaining $n - k$ elements of \mathbf{a} .

4 CVC-STAR: Union of Line Segments

In this section we present a variation of CVC, called CVC-STAR, that is safe against the threat of insider data information but is vulnerable to insider algorithm information. In CVC-STAR the set Π is not a polytope, or even convex. Instead it is the union of n line segments in n -space, which can be thought of as resembling a star. In particular

$$\Pi = \bigcup_{i \in N} S_i \quad (6)$$

and

$$S_i = \{\mathbf{a} - (a_i - \alpha u_i - (1 - \alpha)\ell_i)\mathbf{e}_i : \alpha \in [0, 1]\} \quad (7)$$

where \mathbf{e}_i is the i^{th} unit n -vector. In words S_i is the line segment in which all elements of \mathbf{a} except for a_i retain their original values, while the i^{th} element takes on all values in the range $[\ell_i, u_i]$. It follows that for any query q corresponding to the function f and the set T , that the answer interval can be computed by minimizing and maximizing f over all S_i , $i \in T$ and then concatenating the t answer intervals. Although each set S_i contains only one variable, so that minimization and maximization of the query functions over each S_i should be simple, straightforward application to the solution of (3), (4) for CVC-STAR may still be unwieldy if t is large. Therefore exact algorithms all of which are $o(t)$ and minimal access are developed in Sections 4.2 - 4.6 for a number of standard query types.

Although CVC-POL and CVC-STAR are closely related, the different definitions of Π affect all of the primary considerations of CVC very strongly. Since every answer from CVC-STAR is computed based on $t - 1$ of the elements of \mathbf{a} , it seems logical that CVC-STAR would, in general, produce

tighter intervals than CVC-POL, especially as t increases. In fact the computational experience of Section 6 bears out that intuition. It should also be reemphasized that CVC-POL can give exact answers to some queries, and Π can even be constructed so that this happens with some regularity. A small example of this is given below where $\lambda_1 = .2$, $\lambda_2 = .3$ and a SUM query would be answered with $[120, 120]$.

Record	P ¹	P ²	P ³	a
1	50	70	58	60
2	70	50	62	60

On the other hand exact answers will almost never occur with CVC-STAR. For instance, ignoring \mathbf{P}^3 in the above table, the same query would yield $[110, 130]$. In general, for $x_i \in [\ell_i, u_i]$, CVC-STAR will give an exact answer to a differentiable query function f if $\frac{df}{dx_i} = 0$ for all i and $x_i \in [\ell_i, u_i]$. That condition will only be satisfied in very rare circumstances. For other instances where exact answers are possible consider, the SUM query from Table 3 with $T = \{14\}$. It would be answered with 51 since the subject requires no protection. Also a nondifferentiable query like “The number of subjects with salary greater than 150” would be answered with zero.

4.1 Threat Analysis for CVC-STAR

On the surface it would appear, again because it makes more direct use of elements of \mathbf{a} , that CVC-STAR is less secure than CVC-POL. That turns out to be true with respect to insider algorithm information but not to insider data information. In fact, it is easy to show that, in some sense CVC-STAR provides the ultimate protection against the latter. Observe that CVC-STAR protects against unauthorized disclosure even in the case when all other subjects in the database collude in an attempt to discover a_i . That is, all but a single element a_w of $\{a_i : i \in T\}$ are known to the user asking a query. Even so, there is no query or set of queries they can ask that will reveal more than $a_w \in [\ell_w, u_w]$ even if (3), (4) are solved exactly.

But, suppose that a user has no insider data information, but knows the CVC-STAR algorithm process. That is, the user knows that CVC-STAR gives exact solutions to (3), (4) over Π as defined by (6), (7). The parameters of CVC-STAR are only the vector pair (\mathbf{l}, \mathbf{u}) , all of which the user can also obtain by asking the SUM queries with $T = \{i\}$ for $i = 1, \dots, n$. Consider the following table.

record	\mathbf{l}	a	\mathbf{u}
1	30	50	80
2	20	40	70

A user asks the three SUM queries corresponding to $T = \{1\}$, $\{2\}$, and $\{1, 2\}$, which are answered by CVC-STAR with $[30, 80]$, $[20, 70]$, and $[70, 120]$ respectively. Then the user can immediately observe that the following inequalities hold: $a_1 \geq 50$, $a_2 \geq 40$, $a_1 \leq 50$, $a_2 \leq 40$. Thus, not only are a_1 and a_2 not bound protected but they can be determined exactly.

Thus CVC-POL and CVC-STAR have symmetric strengths and weaknesses. In the remainder of this section we show how to solve (3), (4) exactly by CVC-STAR for some standard query types. Not surprisingly, exact solution of queries is generally simpler for CVC-STAR than for CVC-POL and thus there is never a need to resort to heuristics. All examples except those for REGRESSION queries are illustrated with the data (ignoring \mathbf{P}^3) of Table 2 and compared to the corresponding answers from CVC-POL. The observation that answers for CVC-STAR are generally superior to those of CVC-POL does not become clear from these very small examples as opposed to the results of Section 6.

4.2 MEAN(SUM) Queries:

Let $\Delta_i^+ = u_i - a_i$ and $\Delta_i^- = \ell_i - a_i$. Clearly, for the MEAN (or equivalently SUM) query (3), (4) are solved by

$$f^- = \frac{1}{t} \left(\sum_{i \in T} a_i + \min\{\Delta_i^- : i \in T\} \right)$$

and

$$f^+ = \frac{1}{t} \left(\sum_{i \in T} a_i + \max\{\Delta_i^+ : i \in T\} \right).$$

For instance the query “Mean of the salaries of all employees of Company B” has $T = \{2, 4, 5, 9, 11\}$ and would be answered (ignoring \mathbf{P}^3), with $[43.8, 46.4]$

since $\sum_{i \in T} a_i = 229$, $\min\{\Delta_i^- : i \in T\} = \Delta_5^- = -10$, and $\max\{\Delta_i^+ : i \in T\} = \Delta_4^+ = 3$, so that from $e = 45.8$, $h = .0568$. Note that for SUM and MEAN queries CVC-POL also gives exact solutions to (3), (4) by simply minimizing and maximizing f over the three extreme vectors. For CVC-POL the corresponding answer is $I = [44.2, 46.8]$, with the identical value $h = .0568$.

4.3 MIN Queries:

Given the symmetry between MIN and MAX queries we focus only on the former. Clearly $f^- = \min\{\ell_i : i \in T\}$. To determine f^+ we first observe that for all S_i , f is maximized over S_i at a point having $x_i = u_i$. Then renumber the rows of T in non decreasing order of a_i as $\{w(1), \dots, w(t)\}$. Let $f_{w(1)}^+ := \max_{x \in S_{w(1)}} \min\{x_i : i \in T\}$. From the construction of S_i it follows that

$$f_{w(1)}^+ = \min\{u_{w(1)}, a_{w(2)}\} \quad (8)$$

But from (8) $f_{w(1)}^+ \leq a_{w(2)}$, and since it is clear that $f_{w(1)}^+ \geq a_{w(2)}$ for all $i \in T \setminus w(1)$, it follows that $f^+ = f_{w(1)}^+$ as given by (8). For instance consider the query “minimum of the salaries of all employees of Company B”, having $e = 28$. Then $w(1) = 4$, $\ell_1 = 28$, $w(2) = 2$, $u_4 = 31$, $a_2 = 31$, so that the query is answered with $[28, 31]$, with $h = .107$. For CVC-POL, (3) is solved exactly but a heuristic is used for (4), resulting in the answer interval for the same query of $[26.2, 29.8]$ with $h = .129$.

4.4 PERCENTILE Queries

The results in Section 4.3 are easily generalized to PERCENTILE queries. A p^{th} PERCENTILE query, where $p \in [0, 1]$ is defined by first reindexing the a_i 's for $i \in T$ in nondecreasing order. Then if i^* given by

$$i^* = p(t - 1) + 1$$

is integer, $e = a_{w(i^*)}$. Again, assume the elements of T have been renumbered as in Section 4.3. For expository purposes we treat the special case where i^*

is integer since otherwise a solution is found by simple linear interpolation. MIN, MAX, and MEDIAN are the special cases with $p = 0, 1, .5$ respectively. Assume $i^* \notin \{1, t\}$ since otherwise the query is MIN or MAX. Following the same logic as in Section 4.3, to compute f^- we only have to consider $S_{w(i^*)}, \dots, S_t$ and in each case only at ℓ_i , since clearly f is minimized over S_i at ℓ_i . The reverse is true when computing f^+ . Then let $\ell^* = \min\{\ell_i, i = w(i^*), \dots, t\}$, and $u^* = \max\{u_i, i = 1, \dots, w(i^*)\}$. Then

$$f^- = \max\{\ell^*, a_{w(i^*-1)}\} \quad (9)$$

and

$$f^+ = \min\{u^*, a_{w(i^*+1)}\} \quad (10)$$

Note that (9), (10) hold unless $i^* \in \{1, t\}$, in which case they are undefined. For instance the MEDIAN query, corresponding to $p = 0.5$, $e = 47$ and (9), (10) yields the interval $[46, 50]$ so that $h = .0851$. For CVC-POL, (3) and (4) are both solved heuristically, resulting in the answer interval for the same query of $[45.6, 50]$ with $h = .0936$.

4.5 VARIANCE Queries:

Consider a VARIANCE (or equivalently STANDARD DEVIATION) query of the form $f = \sum_{i \in T} (x_i - \bar{x})^2 / t$. For any $k \in T$ let $x_k := a_k + \Delta_k$, where $\Delta_k \in [\Delta_k^-, \Delta_k^+]$. Then it follows that

$$\frac{\Delta f}{\Delta_k} = \Delta_k \left(\frac{t-1}{t} \Delta_k + 2(a_k - \bar{x}) \right) \quad (11)$$

which is convex in Δ_k and thus maximized at one of the endpoints of the k^{th} line segment. It is minimized at the mean of the other t elements of \mathbf{a} unless that point is outside $[\ell_k, u_k]$ in which case it is also minimized at ℓ_k or u_k .

Then $[f^-, f^+]$ is determined by simply enumerating at most three points on each line segment and evaluating (11). For our running example $e = 206.96$ and the resulting CVC-STAR interval is $[154.16, 225.44]$ so that $h = .3444$. For CVC-POL, (3) and (4) are both solved exactly, resulting in the answer interval for the same query of $[174.2, 232.6]$ with $h = .2821$.

4.5.1 SIMPLE LINEAR REGRESSION Queries

We consider a simple linear regression between the confidential attribute and a numeric non-confidential attribute. The threat from regression arises from a user attempting to utilize the available non-confidential data and the regression results to infer the confidential attribute values. Typically, the users are interested in obtaining the intercept (b_0) and slope (b_1) of the regression line, as well as the R^2 and F statistics.

The Regression Line

The slope and intercept of the optimal regression line $\hat{y} = b_0 + b_1x$, are given by.

$$b_1 = \frac{t \sum_{i \in T} x_i y_i - \sum_{i \in T} x_i \sum_{i \in T} y_i}{t \sum_{i \in T} x_i^2 - (\sum_{i \in T} x_i)^2} \quad (12)$$

$$b_0 = \frac{1}{t} \left(\sum_{i \in T} y_i - b_1 \sum_{i \in T} x_i \right) \quad (13)$$

where we change notation slightly to be consistent with the standard tradition that has x represent the data, so that we now have \mathbf{y} represent an element of Π . Now consider $f(\mathbf{y}) - f(\mathbf{a})$ for $y \in S_i$ and f is given by (12). Letting $y_k = a_k + \Delta_k$, for $k \in T$ and $\Delta_k \in [\Delta_k^-, \Delta_k^+]$, from (12) we observe that

$$\frac{\Delta f}{\Delta_k} = \Delta_k \left(\frac{tx_k - \sum_{i \in T} x_i}{t \sum_{i \in T} x_i^2 - (\sum_{i \in T} x_i)^2} \right)$$

which is linear in Δ_k so that that Δf is maximized at $y_k = u_k$ and minimized at $y_k = \ell_k$ if $\sum_{i \in T} x_i \leq tx_k$, while the reverse holds if $\sum_{i \in T} x_i > tx_k$.

From (13) b_0 is also linear in Δ_k and so is either maximized at $y_k = u_k$ and minimized at $y_k = \ell_k$ or the reverse.

The R^2 statistic

The R^2 statistic (or more precisely R) can be represented, for $\mathbf{y} = \mathbf{a}$, by

$$R = \frac{\sum_{i \in T} x_i a_i - \frac{1}{t} \sum_{i \in T} x_i \sum_{i \in T} a_i}{\left(\sum_{i \in T} x_i^2 - \frac{1}{t} \left(\sum_{i \in T} x_i \right)^2 \right)^{\frac{1}{2}} \cdot \left(\sum_{i \in T} a_i^2 - \frac{1}{t} \left(\sum_{i \in T} a_i \right)^2 \right)^{\frac{1}{2}}}. \quad (14)$$

From (14) we get that for any $y_k = a_k + \Delta_k$, for $k \in T$ and $\Delta_k \in [\Delta_k^-, \Delta_k^+]$,

$$f(\Delta_k) = \frac{\sum_{i \in T} x_i a_i + x_k \Delta_k - \frac{1}{t} \left(\sum_{i \in T} x_i \sum_{i \in T} a_i + \Delta_k \sum_{i \in T} x_i \right)}{\left(\sum_{i \in T} x_i^2 - \frac{1}{t} \left(\sum_{i \in T} x_i \right)^2 \right)^{\frac{1}{2}} \cdot \left(\Delta_k^2 + 2\Delta_k a_k + \sum_{i \in T} a_i^2 - \frac{1}{t} \left(\Delta_k + \sum_{i \in T} a_i \right)^2 \right)^{\frac{1}{2}}}. \quad (15)$$

and from (15) $\frac{df(\Delta_k)}{d\Delta_k} = 0$ yields

$$\Delta_k^* = \frac{\left(\sum_{i \in T} x_i a_i - \frac{1}{t} \sum_{i \in T} x_i \sum_{i \in T} a_i \right) \left(a_k - \frac{1}{t} \sum_{i \in T} a_i \right) - \left(\sum_{i \in T} a_i^2 - \frac{1}{t} \left(\sum_{i \in T} a_i \right)^2 \right) \left(x_k - \frac{1}{t} \sum_{i \in T} x_i \right)}{\left(a_k - \frac{1}{t} \sum_{i \in T} a_i \right) \left(x_k - \frac{1}{t} \sum_{i \in T} x_i \right) - \left(\frac{t-1}{t} \right) \left(\sum_{i \in T} x_i a_i - \frac{1}{t} \sum_{i \in T} x_i \sum_{i \in T} a_i \right)}. \quad (16)$$

It follows that for each $k \in T$, (16) can be evaluated and the resulting Δ_k^* is feasible if it falls in $[\Delta_k^-, \Delta_k^+]$. If not it is discarded and (15) is evaluated only at $\Delta_k \in \{\Delta_k^-, \Delta_k^+\}$ to find the minimum and maximum. If so evaluation of (15) includes Δ_k^* . Note that the F statistic is maximized and minimized at the same points as R^2 .

Results

We illustrate the results with the example data in Table 3 taken from Gopal et. al (2002). Tables 4, 5, and 6 report the regression results using the independent variables x^1 , x^2 , and x^3 corresponding to low, medium, and high correlation respectively.

Actual	l_i	u_i	x^1	x^2	x^3
276.26	275.21	302.84	139.34	136.3	162.41
35.39	32.7	36.24	18.26	291.59	104.81
37.38	37.37	41.11	19.04	107.78	133.84
94.21	92.41	101.83	52.32	137.78	171.32
323.02	308.77	341.08	177.53	364.04	247.57
24.56	22.77	25.23	13.63	119.02	381.73
226.56	216.24	238.9	135.66	252.58	232.39
77.09	70.57	78.28	43.52	110.16	328.45
21.29	20.29	22.42	10.8	54.24	93.09
20.89	20.15	22.24	12.42	102.24	251.75
169.88	155.37	172.35	86.68	178.04	199.64
55.18	50.68	56.2	32.45	56.04	335.04
243.75	226.3	250.67	125.53	258.1	174.22
73.84	72.1	79.49	43.6	128.79	75.11
62.42	62.05	68.29	31.88	74.06	419.99
78.33	71.75	79.58	40.91	106.85	166.26
34.54	32.59	36.05	19.45	75.73	145.17
44.36	40.85	45.29	26.05	103.39	32.37
153	146.07	161.37	78.15	252.48	354.56
31.39	28.27	31.41	17.26	122.2	283.1
180.96	173.5	191.59	100.12	194.68	212.27
56.84	55.26	60.95	29.89	81.44	395.29
142.15	133.47	147.68	78.87	162.07	423.37
37	33.71	37.41	21.78	48.22	259.12
212.31	199.4	220.63	111.38	271.49	370.25

Table 3. Regression Data

	low	actual	high
b_0	94.1443	95.6912	98.2855
b_1	.0474	.0538	.0574
R^2	.00345	.00467	.00544
F	.0798	.1081	.1259
$p - value$.7801	.7453	.7259

Table 4. Regression results for low correlation (x^1)

	low	actual	high
b_0	-11.7969	-9.1530	-7.0668
b_1	.7587	.7762	.7985
R^2	.5091	.5376	.5461
F	23.856	26.74	27.67
$p - value$	$6.22 \cdot 10^{-5}$	$3.1 \cdot 10^{-5}$	$2.45 \cdot 10^{-5}$

Table 5. Regression results for moderate correlation (x^2)

	low	actual	high
b_0	-1.9876	-0.6545	.7797
b_1	1.8280	1.8608	1.9132
R^2	.9852	.9916	.9949
F	1530	2993	3011
$p - value$	$1.51 \cdot 10^{-22}$	$7.3 \cdot 10^{-26}$	$6.8 \cdot 10^{-26}$

Table 6. Regression results for high correlation (\mathbf{x}^3)

The answer intervals around R^2 , F , and the p -value (the probability that there is no linear relationship between the two variables) are quite small and allow the user to deduce that the true relationships are weak, moderate, and strong respectively. The answer intervals around b_1 are also quite small, thus enabling a user to obtain relatively accurate marginal relationship between the non-confidential independent and the confidential dependent attributes. Similar results were obtained with CVC-POL where the algorithms are exact for b_0 , b_1 but grid search is used for R^2 .

4.6 COUNT (SELECT)

The COUNT query asks for the number of records in the database that satisfy the condition C on the confidential attribute. Let $\{L_w, w = 1, \dots, m\}$ be a family of nonempty disjoint intervals defined on the domain of $\{a_i\}$, and $C = \cup_{w=1}^m L_w$. It follows from the definition of S_i that

$$f^- = \begin{cases} e - 1, & \text{if } \exists a_i \in C \text{ with } [\ell_i, u_i] \not\subseteq C \\ e, & \text{otherwise.} \end{cases} \quad (17)$$

and

$$f^+ = \begin{cases} e + 1, & \text{if } \exists a_i \notin C \text{ with } [\ell_i, u_i] \cap C \neq \phi \\ e, & \text{otherwise.} \end{cases} \quad (18)$$

Thus from (17) and (18) the interval $[f^-, f^+]$ will always have range 0, 1, or 2. A SELECT query is identical to the COUNT query except the result is the list of subjects that satisfy the condition C .

For example suppose $C = [20, 30] \cup [50, 70]$. Then $T = \{1, 4, 5, 7, 9, 10, 14\}$, and every S_i , $i \in T$ gives $[r_i^L, r_i^U] = [7]$ except S_2, S_4 which give $[7, 8]$, and S_8, S_{11} which give $[6, 7]$. Thus $e = 7$, $I = [6, 8]$ and $h = 0.286$. The bounds

suggested by (Gopal et al. 2002) for COUNT queries are based on a heuristic that provides $I = [6, 10]$ and $h = 0.572$.

5 Intersection and Union of Disguising Sets

5.1 An Overview

Based on the fact that sets Π that differ in their basic structure or in their parameters offer distinct advantages and disadvantages, one is motivated to consider combining them in various ways involving intersection and/or union. Intuitively intersections of camouflaging sets can potentially provide better answers than any of the original sets, while their union can possibly increase protection at the cost of degradation of answers. For ease of notation these concepts are illustrated for the case in which there are only two original sets, but extension to more than two sets is straightforward. Let I_k be the answer interval to a given query based on a disguising set Π_k that satisfies (5) with $\Pi = \Pi_k$.

Set intersection: $\Pi_a = \Pi_1 \cap \Pi_2$. It follows that $I_a \subseteq I_1$ and $I_a \subseteq I_2$, so that potentially better answers can be provided. On the other hand Π_a will not, in general, satisfy (5). Thus special care must be taken in the choice of Π_1 and Π_2 to make their intersection safe. Another possible concern with set intersection is that solution of (3), (4) may be made more difficult by the intersection process. A possible remedy, if the solution of (3), (4) is relatively easy over Π_1 and Π_2 but difficult over Π_a is to use “answer intersection”. That is, to calculate I_1 and I_2 and then provide the answer $I_1 \cap I_2$. It follows that $I_a \subseteq I_1 \cap I_2$, and therefore that answer intersection is safe as long as Π_a satisfies (5). It will also be shown in Section 5.2 that answer intersection provides a degree of protection beyond that of solution over Π_a .

Set union: $\Pi_b = \Pi_1 \cup \Pi_2$. In this case it is easy to see that concatenation of I_1 and I_2 solves (3), (4) over Π_b . Since the resulting interval can be no better than either I_1 or I_2 , the only reason to consider this option is if additional protection can be provided that is not available for Π_1 or Π_2 . It should also be noted that here it is not necessary that either Π_1 or Π_2 be able to protect every individual, only that every individual be protected by one set or the other.

Based on the above let us consider combining sets within and across the classes CVC-POL and CVC-STAR. First we note that there are an infinite number of possible realizations of the former because its parameters are k and $\lambda_1, \dots, \lambda_k$. On the other hand there is essentially only one realization of CVC-STAR since its parameters \mathbf{l} and \mathbf{u} are assumed to be given. Then we can consider the following possible combination of types.

1. POL \cap POL. This is a valid option that is analyzed in the next subsection.
2. POL \cap STAR. This combination offers no benefits. The intersection of the two sets will simply be another star.
3. POL \cup POL. This combination also offers no benefits since there will be very limited additional protection against algorithmic insider threat over use of a single set from the POL class. In addition the quality of the answers will be degraded.
4. POL \cup STAR. This is of great interest. It offers the possibility of protection when the DBA cannot specify which type of threat is present and is the subject of Section 5.3.

5.2 CVC-INTPOL: Intersection of Polytopes

In this section we consider the “set intersection” option only with respect to polytopes as disguising sets. Then each polytope has to be created so that their intersection satisfies (5). One alternative is to initially create two or more sets as in Section 3, each with $k = 3$. Then the condition that their intersection satisfy (5) can be guaranteed by using the same \mathbf{P}^1 and \mathbf{P}^2 vectors as extreme points for each set but changing the λ vector among sets so that \mathbf{P}^3 varies. That option is illustrated in the example below. A generalization of the idea would be to create the sets so that the same \mathbf{P}^1 and \mathbf{P}^2 , satisfying (5), are found in each polytope but not necessarily at their extreme points.

Clearly the new set Π_a is itself a polytope and, as indicated in Section 5.1, CVC-INTPOL can be expected to provide better answers than those derived from any of the original polytopes. There are downsides as well. First CVC-INTPOL may cost more than CVC-POL since additional vectors that describe the new set will have to be stored and processed in answering queries. Thus, rather than attempt to enumerate the extreme points we are

motivated to use the “answer intersection” option described in Section 5.1 to give heuristic solutions.

In terms of threat analysis, CVC-INTPOL using answer intersection, resembles CVC-POL in that it does not provide protection against insider data information. However it does provides some additional protection against insider algorithmic parameter information protection. The intervals are not produced directly from the intersection polytope. Therefore the user would have to determine the number of polytopes intersected, the number of extreme points of each polytope, and the extreme points themselves before attempting to determine \mathbf{a} as a function of these extreme points. On the other hand, even though answer intersection is used, the DBA should be careful about intersecting too many polytopes since eventually \mathbf{a} will become close to an extreme point of the intersection polytope.

Record	\mathbf{P}^1	\mathbf{P}^2	$\mathbf{P}^3(.2, .3)$	$\mathbf{P}^3(.4, .4)$	\mathbf{a}
1	60	53	54.2	49	55
2	31	29	32.2	35	31
3	99	110	108.4	117	107
4	28	31	26.2	22	28
5	53	64	66.4	62	63
6	78	82	83.6	90	82
7	30	28	29.2	29	29
8	29	31	31.8	31	31
9	63	60	58.8	60	60
10	26	27	27.4	27	27
11	46	50	45.6	47	47
12	34	31	31.8	32	32
13	91	100	91.6	94	94
14	51	51	51	51	51

Table 3

For example Table 3 represents two polytopes with $k = 3$ and \mathbf{P}^1 and \mathbf{P}^2 identical but two different \mathbf{P}^3 vectors where (λ_1, λ_2) associated with each are indicated. The “The mean salary of the employees whose Job is Trainee”, with $T = \{2, 4, 7, 8, 10, 12\}$ has $e = 29.67$, and is answered from $(\lambda_1, \lambda_2) = (.2, .3)$ with $I = [29.5, 29.77]$, so that $h = .00899$. When $(\lambda_1, \lambda_2) = (.4, .4)$ the same query is answered with $I = [29.33, 29.67]$, so that $h = .01124$.

Intersection of the answer intervals results in $I = [29.5, 29.67]$, and $h = .00562$.

5.3 CVC-INTPOLUSTAR

Given that, in essence, CVC-POL and CVC-INTPOL provide protection against algorithmic insider information and that CVC-STAR does the same against data insider information, it is natural to combine these two classes of camouflaging sets in order to gain both kinds of protection. It is easy to see that this can be achieved by simply concatenating the answers from the two set types as suggested in Section 5.1. That is, this option provides protection against a user who has one type of insider information or the other.

CVC-POLUSTAR may not protect against a user who has both insider data and algorithm information. In particular, it is vulnerable if the user has insider data information, is knowledgeable about the algorithms CVC-POL and CVC-STAR, knows the value of k used in CVC-POL, and has access to \mathbf{l} and \mathbf{u} vectors. To strengthen protection against simultaneous knowledge of both types of insider information, the DBA could consider CVC-INTPOLUSTAR. In designing the CVC-INTPOL portion of CVC-INTPOLUSTAR, the DBA can consider the following strategies.

- (a) Use a number of CVC polytopes, each with a different value of k .
- (b) Not all polytopes used in CVC-INTPOL need to provide interval protection to the subjects. It is not even necessary for a polytope to contain the confidential data vector \mathbf{a} .
- (c) Vary the realization of CVC-INTPOL. The DBA, for example, can consider using different sets of polytopes for different query types or query cardinalities.

While these strategies increase the storage and computational burden on the database system, they make the database more robust against insider information threats and can also enhance the quality of the query answers provided to the users.

6 Computational Experience

In this section we present results of extensive numerical experiments that were designed to evaluate the relative performance of CVC-STAR, CVC-INTPOL, and CVC-INTPOLUSTAR. The results provide insights on the

impact of insider data information and insider algorithm information on the quality of the answers. A database containing 1,000 records, five non-confidential attributes A_1, \dots, A_5 , and one confidential numeric attribute was created. The confidential attribute values were drawn from a lognormal distribution with parameters $\delta = 1/3$ and $\gamma = -4/3$, where the random variable is defined by $U = \gamma + \delta \ln |X|$ and $X \sim N(0, 1)$. The values so generated were distributed in the range [20, 980]. Note that lognormal data sets are frequently used since numerical data in organizations often follow this distribution (Neter and Loebbecke (1975), Muralidhar et al. (1995)).

The four non-confidential attributes A_1, \dots, A_4 are categorical. A_1 and A_2 are binary random variables with parameter $p = .3$ and $p = .7$ respectively. A_3 and A_4 are discrete uniform variables over the ranges [1,5] and [1,10] respectively. The data for the attribute A_5 is generated from a normal distribution with parameters $\mu = 50$ and $\sigma = 10$.

The protection levels 10%, 20%, and 50% were considered. That is, for example at the 10% protection level, the values for l_i and u_i were randomly selected such that $l_i \leq a_i \leq u_i$ and $u_i - l_i = .1a_i$ for $i = 1, \dots, 1000$. Two polytopes were created for CVC-POL, each with $k = 3$. The first polytope was created with $\lambda = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$ and the second with $\lambda = (\frac{2}{5}, \frac{2}{5}, \frac{1}{5})$. The intersection of these two polytopes was used to evaluate CVC-INTPOL.

The query sets were generated using the values of the non-confidential fields $A_1 - A_5$. Each query set was created from a unique combination of the first four discrete attributes and values of A_5 depending on whether or not $A_5 \leq 50$. A total of 574 query sets were generated with cardinalities varying from two to 52 records. We intentionally chose small cardinality queries since all realizations of CVC perform generally well for large cardinality queries, and further, small cardinality queries are the natural starting points to attack the database with insider data and algorithm information. The cardinality distribution of the query set is shown below.

Cardinality	% of Query Sets
2-10	61%
11-20	27%
21-52	12%

SUM, STANDARD DEVIATION and MIN queries over the confidential attribute were created from each query set, and were evaluated for each approach.

6.1 Results

Figures 1, 2, and 3 report the improvement in the query answers with CVC-INTPOL over CVC-POL with $\lambda = (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$. The improvement in the query answers were evaluated with the metric $\frac{h(\text{CVC-POL}) - h(\text{CVC-INTPOL})}{h(\text{CVC-POL})}$. Significant improvements were observed for SUM and STANDARD DEVIATION queries. Performance improvements were lower for MIN queries, especially for higher protection levels. At the 50% protection level over 60% of the answers were identical for CVC-INTPOL and CVC-POL. This could partially be attributed to the fact that a heuristic technique was used to compute the MIN query answers.

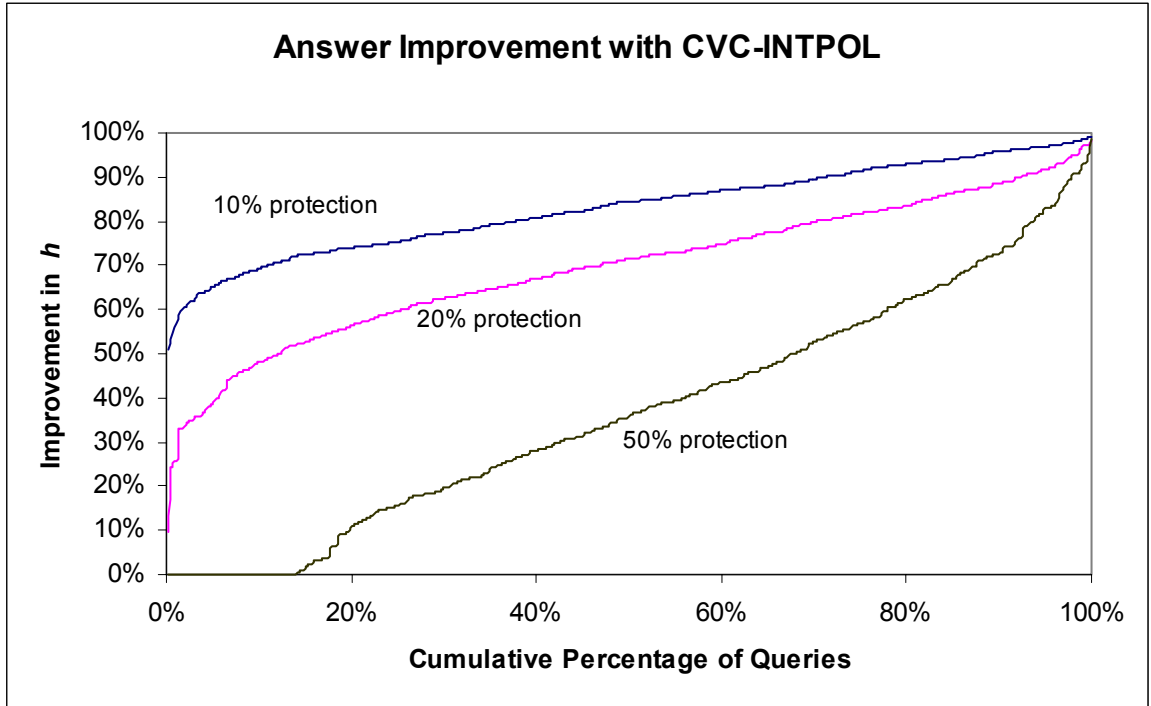


Figure 1: Sum Queries

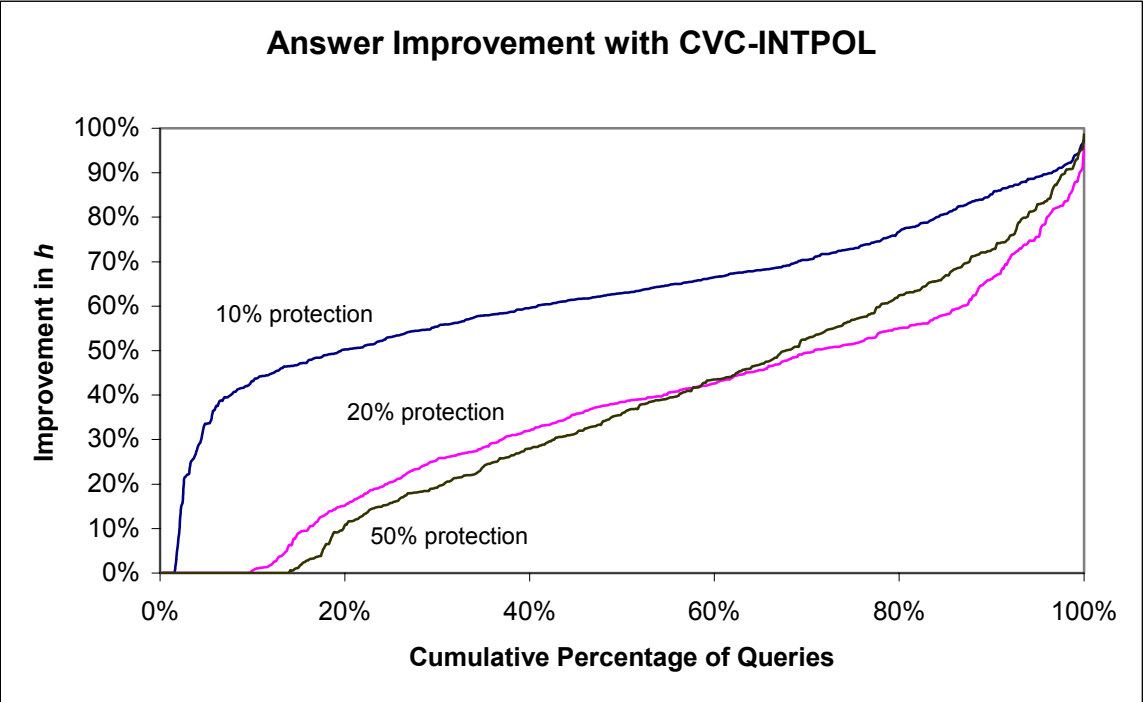


Figure 2: STANDARD DEVIATION QUERIES

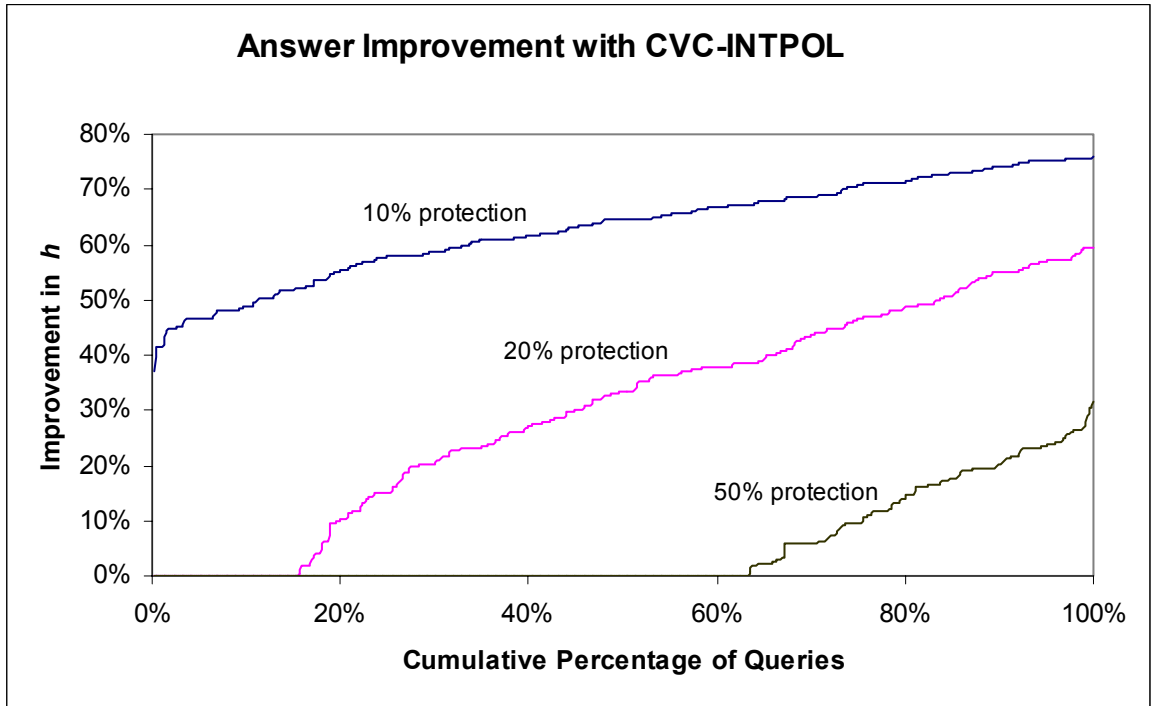


Figure 3: MIN Queries

The average percentage reduction in the answer intervals is reported below.

Protection Level	SUM	STANDARD DEVIATION	MIN
10%	83%	62%	63%
20%	69.5%	36.8%	30%
50%	37%	36.9%	5%

Protection level	Cardinality	SUM		STANDARD DEVIATION		MIN	
		% Dominated by	% Dominated by	% Dominated by	% Dominated by	% Dominated by	% Dominated by
		CVC-STAR	CVC-INTPOL	CVC-STAR	CVC-INTPOL	CVC-STAR	CVC-INTPOL
10%	all queries	68.10%	31.90%	69.20%	30.80%	57.80%	0%
	2-9	68.60%	31.10%	68.20%	31.80%	55%	0%
	>9	67.60%	32.80%	70.30%	29.70%	61.30%	0%
20%	all queries	67.60%	32.40%	69.70%	30.30%	58.30%	0%
	2-9	64.50%	35.20%	68.90%	31.10%	53.80%	0%
	>9	71.50%	28.90%	70.70%	29.30%	64%	0%
50%	all queries	68.80%	31.20%	76%	24%	79.10%	0%
	2-9	69.20%	30.50%	76.40%	23.60%	72.60%	0%
	>9	68.40%	32%	75%	25%	87.10%	0%

Table 7: Relative Performance of CVC-STAR and CVC-INTPOL

In general, the most significant improvements were observed for the 10% protection level. The improvements in the answers was higher for the 20% protection level than for the 50% protection level, except for STANDARD DEVIATION where the average improvements were nearly the same. In fact, for 41% of the queries the reduction in the query answer intervals was higher with the 50% protection level than with the 20% protection level (see Figure 2).

We next compare the performance of CVC-STAR against CVC-INTPOL. Table 7 reports the percentage of the queries for which each of the two techniques provided a better answer. The results indicate that in general CVC-STAR outperforms CVC-INTPOL for all query types. Interestingly, for MIN queries CVC-INTPOL never dominates CVC-STAR. However, for nearly 40% of queries for the 10% and 20% protection levels and 20% of the queries for the 50% level, both techniques provide identical answers. This observation can partly be attributed to the heuristic strategy implemented to evaluate MIN queries in CVC-POL. The performance advantage of CVC-STAR is more dramatic over CVC-POL as illustrated in Table 8.

Protection level	Cardinality	SUM	STANDARD DEVIATION	MIN
10%	all queries	100%	99%	100%
	2-9	100%	98%	100%
	>9	100%	100%	100%
20%	all queries	100%	94.40%	89.60%
	2-9	100%	90.20%	87.40%
	>9	100%	99.60%	92.20%
50%	all queries	93.20%	80%	83.80%
	2-9	90%	78.60%	79.90%
	>9	97.30%	82.40%	88.70%

Table 8: Percentage of Queries for Which CVC-STAR provides strictly better answers than CVC-POL

Table 9 reports the average values of h for CVC-STAR, CVC-INTPOL and CVC-INTPOL \cup STAR. For SUM and STANDARD DEVIATION queries, CVC-INTPOL \cup STAR performs better for larger cardinality queries. For MIN queries, the performance of CVC-INTPOL and CVC-INTPOL \cup STAR are identical since CVC-STAR always performs as well or better than CVC-INTPOL.

Protection level	Cardinality	SUM			STANDARD DEVIATION			MIN		
		CVC-STAR	CVC-INTPOL	CVC-INTPOL \cup STAR	CVC-STAR	CVC-INTPOL	CVC-INTPOL \cup STAR	CVC-STAR	CVC-INTPOL	CVC-INTPOL \cup STAR
10%	all queries	0.04	0.064	0.071	0.11	0.158	0.174	0.098	0.149	0.149
	2-9	0.051	0.076	0.084	0.14	0.19	0.209	0.099	0.148	0.148
	>9	0.027	0.049	0.054	0.074	0.119	0.129	0.097	0.15	0.15
20%	all queries	0.08	0.121	0.135	0.222	0.299	0.341	0.188	0.276	0.276
	2-9	0.101	0.15	0.167	0.282	0.371	0.43	0.191	0.277	0.277
	>9	0.053	0.086	0.095	0.148	0.21	0.232	0.185	0.274	0.274
50%	all queries	0.198	0.318	0.348	0.516	0.721	0.814	0.458	0.705	0.705
	2-9	0.25	0.386	0.422	0.64	0.855	0.979	0.464	0.7	0.7
	>9	0.135	0.234	0.255	0.364	0.556	0.611	0.451	0.711	0.711

Table 9: h Values for CVC-STAR, CVC-INTPOL and CVC – INTPOL \cup STAR

Table 10 presents the relative decrease in answer quality with CVC-INTPOL \cup CVC-STAR over CVC-STAR and CVC-INTPOL. This is captured

with the measures $\frac{h(\text{CVC-STAR}) - h(\text{CVC-INTPOL} \cup \text{STAR})}{h(\text{CVC-STAR})}$ and $\frac{h(\text{CVC-INTPOL}) - h(\text{CVC-INTPOL} \cup \text{STAR})}{h(\text{CVC-INTPOL})}$.

Protection level	Cardinality	SUM		STANDARD DEVIATION		MIN	
		CVC-STAR	CVC-INTPOL	CVC-STAR	CVC-INTPOL	CVC-STAR	CVC-INTPOL
10%	all queries	88.90%	44.20%	67.80%	22.10%	53.00%	0.00%
	2-9	72.00%	43.40%	58.80%	20.20%	51.30%	0.00%
	>9	110.00%	45.00%	79.10%	24.30%	55.00%	0.00%
20%	all queries	78.90%	36.10%	60.70%	23.30%	51.80%	0.00%
	2-9	70.80%	42.60%	62.00%	20.20%	49.50%	0.00%
	>9	89.10%	28.10%	59.00%	27.10%	54.60%	0.00%
50%	all queries	86.00%	40.00%	72.30%	15.10%	58.30%	0.00%
	2-9	72.60%	40.70%	65.90%	16.30%	55.30%	0.00%
	>9	102.60%	39.20%	80.40%	13.70%	62.00%	0.00%

Table 10: Percentage Increase in h with CVC – INTPOL \cup STAR

Performance deterioration is significantly larger if the DBA moves from CVC-STAR to CVC-INTPOL \cup STAR than from CVC-INTPOL to CVC-INTPOL \cup STAR. This result implies that it is more “costly” to protect against insider algorithm information than to provide protection against insider data information.

In summary, our computational results suggest the following: (i) intersection of CVC-POLs can result in significant performance improvements, (ii) the performance of CVC-STAR is significantly better than CVC-POL and even than CVC-INTPOL, and (iii) performance deterioration is significantly higher when the DBA provides protection against insider algorithm information as opposed to protection against insider data information.

7 Conclusions and Future Research

The major thrust of this work has been to categorize two distinct types of insider information that a user of a database may have, and to combat them

while giving deterministically exact answers to queries. The two types of threats are either of knowledge of data in the confidential field or of the algorithmic process that is used to answer queries. The second type has not been considered before in this line of research. We show that a previous realization of the Confidentiality via Camouflage (CVC) technique is safe against the second type but not the first. Thus we develop models and algorithms to combat the first type of threats and combinations of the two to protect against a user who is only known to have one type of knowledge or the other but not which one. Computational experience relates the degradation of answer intervals that can be expected based on the type of threat that is protected against. The general conclusion is that it is less expensive to protect against data knowledge than against process knowledge. Future research, among other extensions, can further explore the proposed methods to protect against a user who poses both types of threats simultaneously.

Acknowledgements

The authors received support from TECI - The Treibick Electronic Commerce Initiative, Department of Operations and Information Management, University of Connecticut. We would also like to thank Dr. Manuel Nunez for his insightful comments and suggestions.

References

- Beck, L. L. 1980. A Security Mechanism for Statistical Databases. *ACM Transactions on Database Systems* **5**, 316-338.
- Chin, F. Y. and G. Ozsoyoglu. 1982. Auditing and Inference Control in Statistical Databases. *IEEE Transactions on Software Engineering* **SE-8**, 574-582.
- Denning, D.E. 1980. Secure Statistical Databases with Random Sample Queries. *ACM Transactions on Database Systems* **5**, 291-315.
- Dobkin, D., A. K. Jones, and R. J. Lipton, 1979. Secure Databases: Protection Against User Influence. *ACM Transactions on Database Systems* **4**, 97-100.
- Duncan, G. T. and S. Mukherjee. 2000. Optimal disclosure limitation strategy in statistical databases: Deterring tracker attacks through additive noise. *Journal of the American Statistical Association* **95**, 720-729.
- Federal Information Processing Standards Publication, Data Encryption Standard (DES), FIPS PUB 46-3, 1999.

- Fellegi, I. P. On the Question of Statistical Confidentiality. 1972. *Journal of the American Statistical Association*, **67**, 7-18.
- Garfinkel, R., R. Gopal and P. Goes. 2002, Privacy Protection of Binary Confidential Data against Deterministic, Stochastic, and Insider Threat, *Management Science* 48, 749-764.
- Gopal, R., R. Garfinkel, and P. Goes. 2002. Confidentiality Via Camouflage: The CVC Approach to Disclosure Limitation when Answering Queries to Databases. *Operations Research*, **50**, 501-516.
- Gopal, R., P. Goes, and R. Garfinkel. 1998. Interval Protection of Confidential Information in a Database. *INFORMS Journal on Computing*, **10**, 309-322.
- Hoffman, L. J., and W. F. Miller. 1970. Getting a Personal Dossier from a Statistical Data Bank. *Datamation* **16**, 74-75.
- Lefons, D., A. Silvestri, and F. Tangorra, 1982. An Analytic Approach to Statistical Databases. *Proceedings of 9th Conference on Very Large Databases*, 189-196.
- Leiss, E. 1982. Randomizing: A Practical Method for Protecting Statistical Databases against Compromise. *Proceedings of 8th Conference on Very Large Databases*.189-196.
- Liew, C. K., W.J. Choi, and C.J. Liew. 1985. A Data Distortion by Probability Distribution. *ACM Transactions on Database Systems* **10**, 395-411.
- Muralidhar, K., D. Batra, and P.J. Kirs. 1995. Accessibility, Security, and Accuracy in Statistical Databases: The Case for the Multiplicative Fixed Data Perturbation Approach. *Management Science* **41**, 1549-1564.
- Netter, J. and J.K. Loebeckke.1975. Behavior of Major Statistical Estimators in Sampling Accounting Populations: An Empirical Study. *American Institute of Certified Public Accountants*.
- Pfleeger, C.P., S.L. Pfleeger, and W.H.Ware 2002. *Security in Computing*. Prentice Hall, New York.
- Reiss, S. P. 1984. Practical Data Swapping: The First Steps. *ACM Transactions on Database Systems* **9**, 20-37.
- Sarathy, R. and K. Muralidhar. 2002. The Security of Confidential Numerical Data in Databases. *Information Systems Research*, **13** 389-403.
- Schlorer, J. 1980. Disclosure from Statistical Databases: Quantitative Aspects of Trackers. *ACM Transactions on Database Systems*. **5** 467-492.
- Schlorer, J. 1981. Security of Statistical Databases: Multidimensional Transformation. *ACM Transactions on Database Systems* **6** 95-112.

Traub, J. F., Y. Yemini, and H. Wozniakowski. 1984. The statistical security of a statistical database." *ACM Transactions On Database Systems* **9** 672-679.