DISCLOSURE CONTROL OF CONFIDENTIAL DATA
BY APPLYING PAC LEARNING THEORY

By

LING HE

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2005

I would like to dedicate this work to my parents, Tianqin He and Yan Gao, for their endless love and encouragement through all these years.

ACKNOWLEDGMENTS

I would like to express my complete gratitude to my advisor, Dr. Gary Koehler. This dissertation would not have been possible without his support, guidance, and encouragement. I have been very fortunate to have an advisor who is always willing to devote his time, patience and expertise to the students. During my Ph.D. program, he taught me invaluable lessons and insights on the workings of academic research. As a distinguished scholar and a great person, he sets an example that always encourages me to seek excellence in the academic area as well as my personal life.

I am very grateful to my dissertation cochair, Dr. Haldun Aytug. His advice, support and help in various aspects of my research carried me on through a lot of difficult times. In addition, I would like to thank the rest of my thesis committee members: Dr. Selwyn Piramuthu and Dr. Anand Rangarajan. Their valuable feedback and comments helped me to improve the dissertation in many ways.

I would also like to acknowledge all the faculty members in my department, especially the department chair, Dr. Asoo Vakharia, for their support, help and patience.

I also thank my friends for their generous help, understanding and friendship in the past years. My thanks also go to my colleagues in the Ph.D. program for their precious moral support and encouragement.

Last, but not least, I would like to thank my parents for always believing in me.

TABLE OF CONTENTS

page

CHAPTER

APPENDIX

LIST OF TABLES

LIST OF FIGURES

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

DISCLOSURE CONTROL OF CONFIDENTIAL DATA BY APPLYING PAC LEARNING THEORY

By

Ling He

August 2005

Chair: Gary Koehler
Cochair: Haldun Aytug
Major Department: Decision and Information Sciences

With the rapid development of information technology, massive data collection is relatively easier and cheaper than ever before. Thus, the efficient and safe exchange of information becomes the renewed focus of database management as a pervasive issue. The challenge we face today is to provide users with reliable and useful data while protecting the privacy of confidential information contained in the database.

Our research concentrates on statistical databases, which usually store a large number of data records and are open to the public where users are allowed to ask only limited types of queries, such as Sum, Count and Mean. Responses for those queries are aggregate statistics that intends to prevent disclosing the identity of a unique record in the database.

My dissertation aims to analyze these problems from a new perspective using Probably Approximately Correct (PAC) learning theory which attempts to discover the true function by learning from examples. Different from traditional methods from which database administrators apply security methods to protect the privacy of statistical databases, we regard the true database as the target concept that an adversary tries to discover using a limited number of queries, in the presence of some systematic perturbations of the true answer. We extend previous work and classify a new data perturbation method– the variable data perturbation which protects the database by adding random noises to the confidential field. This method uses a parametrically driven algorithm that can be viewed as generating random perturbations by some (unknown) discrete distribution with known parameters, such as the mean and standard deviation. The bounds we derive for this new method shows how much protection is necessary to prevent the adversary from discovering the database with high probability at small error. Put in PAC learning terms we derive bounds on the amount of error an adversary makes given a general perturbation scheme, number of queries and a confidence level.

CHAPTER 1
INTRODUCTION

1.1 Background

Statistical organizations, such as U.S. Census Bureau, National Statistical Offices (NSOs), and Eurostat, collect large amounts of data every year by conducting different types of surveys from assorted individuals. Meanwhile, the data stored in the statistical databases (SDBs) are disseminated to the public in various forms, including microdata files, tabular data files or sequential queries to the online databases. The data are retrieved, summarized and analyzed by various database users, i.e., researchers, medical institutions or business companies. Among the published data, restrictions are established on the release of sensitive data in order to comply with the confidentiality agreements imposed by the sources or providers of the original information. Therefore, the protection of confidential information becomes a critical issue with serious economic and legal implications which in turn expands the scope and necessity of improved security in the database field.

Statistical databases usually store large a number of data records and are open to the public where users are allowed to ask only limited types of queries, such as Sum, Count and Mean. Responses for those queries are aggregate statistics that aim to prevent disclosing the identity of a unique record in the database.

With the rapid development of information technology, it becomes relatively easier and cheaper to obtain data than ever before. With the recent passage of The Personal Responsibility and Work Opportunity Act of 1996 (The Welfare Reform Act) (Fiengerg

2000) and Health Insurance Portability and Accountability Act of 1996 (HIPPA) in the United States, the protection of confidential information collected by statistical organizations has become a renewed focus of database management as a pervasive issue since the 70s and 80s. Those statistical organizations have the legal and ethical obligations to maintain the accuracy, integrity and privacy of the information contained in their databases.

1.2 Motivation

Traditional research on SDBs privacy, which is also called Statistical Disclosure Control (SDC), has been under way for over 30 years. SDC provides all types of security-control methods. Among them, microaggregation, cell suppression and random data perturbation are some of the most promising SDC methods. Recently, Garfinkel et al. (2002) developed a new technique called CVC protection which designs a network algorithm to construct a series of camouflage vectors which hides the true confidential vector. This CVC technique provides interval answers to ad-hoc queries. All those SDC methods attempt to provide the SDB users with reliable and useful data (minimizing the information loss) while protecting the privacy of the confidential information in the database (minimizing the disclosure risk) as well.

Probably Approximately Correct (PAC) learning theory is a framework for analyzing machine learning algorithms. It attempts to discover the true function by learning from examples which are randomly drawn from an unknown but fixed distribution. Given accuracy and confidence parameters, the PAC model bounds the error that the true function makes.

Different from the traditional methods from which database administrators apply SDC methods to protect the privacy of SDBs, we approach the database security problem from a new perspective, from which we assume that an adversary regards the true confidential data in the database as the target concept and tries to discover it within a limited number of queries by applying PAC learning theory.

We describe how much protection is necessary to guarantee that the adversary cannot uncover the database's confidential information with high probability. Put in PAC learning terms we derive bounds on the amount of error an adversary makes given a general perturbation scheme, number of queries and a confidence level.

1.3 Research Problem

Additive data perturbation includes some of the most popular database security methods. Inspired by the CVC technique, we classify a new method into this category– the variable data perturbation which protects a database by adding random noises. Different from the fixed random data perturbation method, this method effectively generates random perturbations which have an unknown discrete distribution. However, parameters, such as the mean and standard deviation, can be estimated. The variable data perturbation method is the focus of our research.

We intend to derive a bound on the level of error that an adversary may make while compromising a database. We extend the previous work by Dinur and Nissim (2003), who found a bound for the fixed data perturbation method, and deploy the PAC learning theory to develop a new bound for the variable data perturbation.

A threshold on the number of queries is developed from the error bound. With high probability, the adversary can disclose the database at small error if this certain number of queries is asked. Therefore, we may find out how much protection would be necessary to prevent the disclosure of the confidential information in a statistical database.

Our experiments indicate that a high level of protection may yield answers that are not useful whereas useful answers can lead to the compromise of a database.

1.4 Contribution

Two major contributions are expected from this research. First, we approach the database security problem from a new perspective instead of following the traditional research paths in this field. By applying PAC learning theory, we regard an adversary of the database as a learner who tries to discover the confidential information within a certain number of queries. We show that both SDC methods and PAC learning theory actually use the similar methodology for different purposes. We also derive a PAC-like bound on the sample size for the variable data perturbation method, within which the database can be compromised with a high probability at small error. Based on this result, we would find out if a security method can provide enough protection to the database.

1.5 Organization of Dissertation

The dissertation is organized into 8 parts. Chapter 2 provides an overview of the important concepts, methodologies and models in the fields of machine learning and PAC learning theory. In Chapter 3, we summarize database security-control methods in microdata files, tabular data files and the statistical database which is the emphasis of our efforts. We review the literature of performance measurements for the database protection methods in Chapter 4. Following that, in Chapter 5 random data perturbation methods are reviewed and a new data perturbation method, variable-data perturbation, is defined and developed. Two papers that motivated our research are reviewed and explained. We propose our approach at the end of this chapter. In Chapter 6, we introduce our methodology and develop the research model. A bound on the sample size for the variable data perturbation method is derived, within which the confidential information can be disclosed. In Chapter 7, experiments are designed and conducted to test our theoretical conclusions from previous chapters. Experimental results are summarized and analyzed at the end. Chapter 8 concludes our work and gives directions for future research.

CHAPTER 2
STATISTICAL AND COMPUTATIONAL LEARNING THEORY

In this chapter, we introduce Statistical and Computational Learning Theory, a formal mathematical model of learning. The overview focuses on the PAC model, the most commonly used theoretical framework in this area. We then move to a brief review of statistical learning theory and its two important principles: empirical and structural minimization principles. Other well-known concepts and theorems are also investigated here. At the end of the chapter, we extend the basic PAC framework to more practical models, that is, learning with noise and query learning models.

2.1 Introduction

Since the 1960s, researchers have been diligently working on how to make computing machines learn. Research has focused on both empirical and theoretical approaches. The area is now called machine learning in computer science but referred to as data mining, knowledge discovery, or pattern recognition in other disciplines. Machine learning is a mainstream of artificial intelligence. It aims to design learning algorithms that identify a target object automatically without human involvement. In the machine learning area, it is very common to measure the quality of a learning algorithm based on its performance on a sample dataset. It is therefore difficult to compare two algorithms strictly and rigorously if the criterion depends only on empirical results. Computational learning theory defines a formal mathematical model of learning, and it makes it possible to analyze the efficiency and complexity of learning algorithms at a theoretical level (Goldman 1991).

2.2 Machine Learning

2.2.1 Introduction

In this section we start our review with an introduction to important concepts in the machine learning field, such as hypotheses, training samples, instances, instance spaces, etc. This is followed by a demonstration of the basic machine learning model which is designed to generate an hypothesis that closely approximates the unknown target concept. See Natarajan (1991) for a complete introduction.

2.2.2 Machine Learning Model

Many machine learning algorithms are utilized to tackle classification problems which attempt to classify objects into particular classes. Three types of classification problems includ binary classification–one with two classes; multi-class classification–handling a finite number of output categories; and regression whose output are real values (Cristianini and Shawe-Taylor 2000).

Most machine learning methods learn from examples of the target concept. This is called supervised learning. The *target concept (or target function)* f is an underlying function that maps data from the input space to the output space. The input space is also called an instance space, denoted as X, which is used to describe each instance $x \in X \subseteq \Re^n$. Here n represents the dimensions or attributes of the input instance. The output space, denoted as Y, contains every possible output label $y \in Y$. In the binary classification case, the target concept (or target function) $f\left(x\right)$ classifies all *instances* $x \in X$ into negative and positive classes, illustrated as 0 and 1, $X \subseteq \Re^n \rightarrow Y \subseteq \left\{0,1\right\}$. Let $f\left(x\right)=1$ if x belongs to a positive (true) class, and $f\left(x\right)=0$ (false) otherwise.

Suppose a sample S includes l pairs of training examples, $S=\left(\left(x_1,y_1\right),\cdots,\left(x_l,y_l\right)\right)$. Each x_i is an instance, and output y_i is x_i's classification label.

The learning algorithm inputs the training sample and outputs an hypothesis $h\left(x\right)$ from the set of all hypotheses under consideration which best approximates the target concept $f\left(x\right)$ according to its criteria. An *hypothesis space* H is a set of all possible hypotheses. The *target concept* is chosen from the *concept space*, $f\in C$, which consists of a set of all possible concepts (functions).

2.3 Probably Approximately Correct Learning Model

2.3.1 Introduction

The PAC model proposed by Valiant in 1984 is considered the first formal theoretical framework to analyze machine learning algorithms, and it formally initiated the field of computational learning theory. By learning from examples, the PAC model combines methods from complexity theory and probability theory, aimed at measuring the complexity of learning algorithms. The core idea is that the hypothesis generated from the learning algorithm approximates the target concept with a high probability at a small error in polynomial time and/or space.

2.3.2 The Basic PAC Model Learning Binary Functions

The PAC learning model quantifies the worst-case risk associated with learning a function. We discuss its details using binary functions as the learning domain. Suppose there is a training sample S of size l. Every example is generated independently and identically from an unknown but fixed probability distribution D over the instance space $X\subseteq\left\{0,1\right\}^n$. Thus, the PAC model is also named a distribution-free model. Each instance is an n -bits binary vector, $x \in X \subseteq \{0,1\}^n$. The learning task is to choose a specific boolean function that approximates the target concept $f : \{0,1\}^n \rightarrow \{0,1\}$, $f \in C$. The *target concept* f is chosen from the *concept space* $C = 2^X$ of all possible boolean functions. According to PAC requirements a learning algorithm must output an hypothesis $h \in H$ in polynomial time, where $H \subseteq 2^X$. We hope that the target function $f \in H$ and hypothesis h can approximate target function f as accurately as possible. If $f \notin H$ then the classification errors are inevitable.

Consider a concept space $C = 2^X$, an hypothesis space $H \subseteq 2^X$, and an unknown but fixed probability distribution D over an instance space $X \subseteq \{0,1\}^n$, the error of an hypothesis, $h \in H$ with respect to a target concept $f \in C$, is the probability that h and f disagree on the classification of an instance $x \in X$ drawn from D. This probability of error is denoted by a risk functional:

$$\underset{D}{err}(h) = \Pr_D \{(x, f(x)) : h(x) \neq f(x)\}$$

To understand the error more intuitively, see Figure 2-1. The error probability is indicated by areas of I and II. Areas I and II in the figure show where $h(x)$ disagrees with $f(x)$ on the instances located in these places. We can think about them as Type I and Type II errors. Area III and IV contain those instances that $h(x)$ and $f(x)$ agree on their classification.

The PAC model utilizes an accuracy parameter ε and confidence parameter δ to measure the quality of an hypothesis h. Given a sample S of size l, and a distribution D

from which all training examples are drawn, the PAC model strives to bound the probability that an hypothesis h gives large error by δ as in

$$\Pr_D^l\left\{S : error_D\left(h_s\right) > \varepsilon\right\} < \delta$$

where h_s means that the training set decides the selection of the hypothesis.

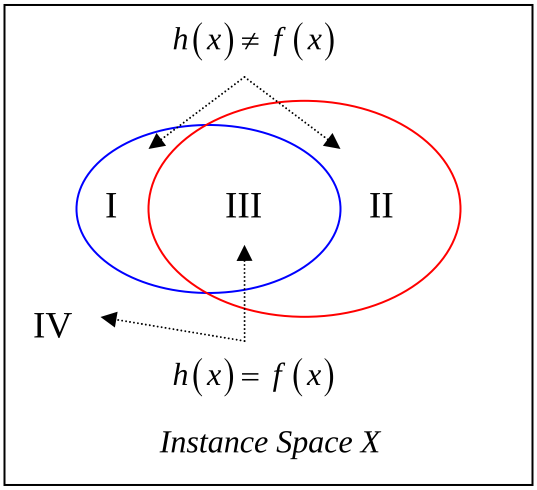


Figure 2-1: Error Probability

Definition: PAC Learnable. A concept class C of boolean functions is *PAC learnable* if there exists a learning algorithm A, using an hypothesis space H, such that for every $f \in C$, for every probability distribution D, for every $0 < \varepsilon < 1/2$, and for every $0 < \delta < 1/2$:

(1) An hypothesis $h \in H$, produced by algorithm A, can approximate the target function f with high probability at least $1-\delta$, such that $error(h) \leq \varepsilon$.

(2) The complexity of the learning algorithm A is bounded by the size of target concept n, $1/\varepsilon$ and $1/\delta$ in polynomial time. The sample complexity refers to the sample size within which the algorithm A needs to output an hypothesis h.

2.3.3 Finite Hypothesis Space

An hypothesis space H can be finite or infinite. If an hypothesis h classifies all training examples correctly, it is called a *consistent hypothesis*. We will derive the main PAC result in multiple steps using well-known inequalities from probability theory.

2.3.3.1 Finite consistent hypothesis space

Assuming the hypothesis space H is finite, if we choose an hypothesis h with a risk greater than ε, the probability that it is consistent on a training sample S of size l is bounded as

$$\Pr_D^l\left\{S: \ h \ consistent \ and \ error(h) > \varepsilon\right\} \le (1-\varepsilon)^l \le e^{-\varepsilon l}.$$

To see this, observe that the probability that hypothesis h_1 classifies one input pair $\left(x_1, f\left(x_1\right)\right)$ correctly is $\Pr^1\left\{h_1\left(x_1\right) = f\left(x_1\right)\right\} \le (1-\varepsilon)$. Given l examples, the probability h_1 classifies $\left(x_1, f\left(x_1\right)\right), \cdots, \left(x_l, f\left(x_l\right)\right)$ correctly is

$$\Pr^l\left\{\left(h_1\left(x_1\right) = f\left(x_1\right)\right) \wedge \cdots \wedge \left(h_l\left(x_l\right) = f\left(x_l\right)\right)\right\} \le (1-\varepsilon)^l$$

because the sampling is i.i.d. Thus, the probability of finding an hypothesis h with error greater than ε and consistent with the training set (of size l) is denoted by the union bound (i.e., the worst case) $|H|(1-\varepsilon)^l$. To see this latter step, first define E_i to represent the event that h_i is consistent. Then we know that

$$\Pr^l\left\{\bigcup_{i=1}^{|H|} E_i\right\} \le \sum_{i=1}^{|H|} \Pr^l\left\{E_i\right\} \le |H|(1-\varepsilon)^l.$$

Finally, $(1-\varepsilon)^l \le e^{-\varepsilon l}$ is a commonly known simple algebraic inequality.

The idea behind the PAC bound is to bound this unlucky scenario (i.e., algorithm A finds a consistent hypothesis that happens to be one with error greater than ε). The following result formalizes this.

Blumer Bound (Blumer et al. 1987). $|H|(1-\varepsilon)^l \leq \delta$. Thus, the sample complexity, l , for a consistent hypothesis h over finite hypothesis space H , is bounded by

$$l \geq \frac{1}{\varepsilon}\left(\ln|H| + \ln\frac{1}{\delta} \right)$$

2.3.3.2 Finite inconsistent hypothesis space

An hypothesis h is called inconsistent if there exist misclassification errors $\varepsilon_s > 0$ in the training sample. The sample complexity is therefore bounded by

$$l \geq \frac{1}{2(\varepsilon - \varepsilon_s)^2}\left(\ln|H| + \ln\frac{1}{\delta} \right)$$

and the error is bounded by

$$\varepsilon \geq \varepsilon_s + \sqrt{\frac{1}{2l}\left(\ln|H| + \ln\frac{1}{\delta} \right)}$$

We can see from the above inequality that ε is usually larger than error rate ε_s . Interested readers can see Goldman (1991) for further explanations.

2.3.4 Infinite hypothesis space

When H is finite we can use $|H|$ directly to bound the sample complexity. When H is infinite we need to utilize a different measure of capacity. One such measure is called the VC dimension, which was first proposed by Vapnik and Chervonenkis (1971).

Definition: VC Dimension Definition. The VC dimension of an hypothesis space is the maximum number, d , of points of the instance space that can be separated into two classes in all possible 2^d ways using functions in the hypothesis space. It measures the richness or capacity of H (i.e., the higher d is the richer the representation). Given *H* with a VC dimension d and a consistent hypothesis $h \in H$ then the PAC error bound is (Cristianini and Shawe-Taylor 2000):

$$\varepsilon \leq \frac{2}{l}\left(d \log_2 \frac{2el}{d} + \log_2 \frac{2}{\delta} \right)$$

provided $d \leq l$ and $l > 2/\varepsilon$.

2.4 Empirical Risk Minimization and Structural Risk Minimization

2.4.1 Empirical Risk Minimization

Given a VC dimension d and an hypothesis $h \in H$ with a training error ε_s, the error rate ε is bounded by

$$\varepsilon < 2\varepsilon_s + \frac{4}{l}\left\{ d \ln \frac{2el}{d} + \ln \frac{4}{\delta} \right\}$$

Therefore, the empirical risk can be minimized directly by minimizing the number of misclassifications on the sample. This principle is called the *Empirical Risk Minimization* principle.

2.4.2 Structural Risk Minimization

As is well known, one disadvantage of the empirical risk minimization is the over-fitting problem, that is, for small sample sizes, a small empirical risk does not guarantee a small overall risk. Statistical learning theory uses the *structural risk minimization principle (SRM)* (Schölkopf and Smola 2001, Vapnik 1998) to solve this problem. The SRM focuses on minimizing a bound on the risk functional.

Minimizing a risk functional is formally developed as a goal of learning a function from examples by statistical learning theory (Vapnik 1998):

$$R(\alpha) = \int L\left(z, g(z,\alpha)\right) dF(z)$$

over $\alpha \in \Lambda$ where $L(\ )$ is a loss function for misclassified points, $g(\bullet,\alpha)$ is an instance of a collection of target functions parametrically defined by $\alpha \in \Lambda$, and z is the training pair assumed to be drawn randomly and independently according to an unknown but fixed probability distribution $F(z)$. Since $F(z)$ is unknown, an induction principle must be invoked.

It has been shown that for any $\alpha \in \Lambda$ with a probability at least $1-\delta$, the bound on a consistent hypothesis

$$R(\alpha) \le R_{emp}(\alpha) + \frac{R_{struct}(d,l,\delta)}{2}\left(1+\sqrt{1+\frac{4R_{emp}(\alpha)}{R_{struct}(d,l,\delta)}}\right) \equiv R_{bound}(\alpha)$$

holds where the structural risk $R_{struct}(\ )$ depends on the sample size, l, the confidence level, δ, and the capacity, d, of the target function. The bound is tight, up to log factors, for some distributions (Cristianini and Shawe-Taylor 2000). When the loss function is the number of misclassifications, the exact form of $R_{struct}(\ )$ is

$$R_{struct}(d,l,\delta) = 4\frac{d\left(\ln(2l/d)+1\right) - \ln(\delta/4)}{l}$$

It is a common learning strategy to find consistent target functions that minimize a bound on the risk functional. This strategy provides the best “worst case” solution, but it does not guarantee finding target functions that actually minimize the true risk functional.

2.5 Learning with Noise

2.5.1 Introduction

The basic PAC model is also called the noise-free model since it assumes that the training set is error-free, meaning that the given training examples are correctly labeled and not corrupted. In order to be more practical in the real world, the PAC algorithm has been extended to account for noisy inputs (defined below). Kearns (1993) initiated another well-studied model in the machine learning area, the Statistical Query model (SQ), which provides a framework for a noise-tolerant learning algorithm.

2.5.2 Types of Noise

Four types of noise are summarized in Sloan's paper (Sloan 1995):

(1) Random Misclassification Noise (RMN)

Random misclassification noise occurs when the learning algorithm, with probability $1-\eta$, receives noiseless samples (x, y) from the oracle and, with probability η, receives noisy samples $(x, \bar{y})$ (i.e., x with an incorrect classification). Angluin and Laird (1988) first theoretically modeled PAC learning with RMN noise. Their model presented a benign form of misclassification noise. They concluded if the rate of misclassification is less than $1/2$, then the true concept can be learned by a polynomial algorithm. Within l number of samples, the algorithm can find an hypothesis h minimizing the number of disagreements $F(h,\sigma)$. Disagreements $F(h,\sigma)$ denotes the number of times that some hypothesis h disagrees with σ, where σ is the training sample. Sample size l is bounded by

$$l \geq \frac{2}{\varepsilon^2 (1-2\eta_b)^2} \ln\left(\frac{2|H|}{\delta}\right)$$

provided $0 < \eta < \eta_b < 1/2$.

Extensive studies can be found in Aslam and Decatur (1993), Blum et al. (1994), Bshouty et al. (2003), Decatur and Gennaro (1995), and Kearns (1993).

(2) Malicious Noise (MN)

Malicious noise occurs when the learning algorithm, with probability $1-\eta$, gets the correct samples but with probability η the oracle returns noisy data, which may be chosen by a powerful malicious adversary. No assumption is made about corrupted data, and the nature of the noise is also unknown. Valiant (1985) first simulated this situation of learning from MN. Kearns and Li (1993) further analyzed this worst-case model of noise and presented some general methods that any learning algorithm can apply to bound the error rate, and they showed that learning with noise problems are equivalent to standard combinatorial optimization problems. Additional work can be found in Bshouty (1998), Cesa-Bianchi et al. (1999), and Decatur (1996, 1997).

(3) Malicious Misclassification Noise (MMN)

Malicious misclassification (labeling) noise is that where misclassification is the only possible noise. The adversary can choose only to change the label y of the sample pair (x, y) with probability η, while no assumption is made about y. Sloan (1988) extended Angluin and Laird's (1988) result to this type of noise.

(4) Random Attribute Noise (RAN)

Random attribute noise is as follows. Suppose the instance space is $\{0,1\}^n$. For every instance x in a sample pair (x, y), its attribute x_i, $1 \le i \le n$, is flipped to $\overline{x}_i$ independently and randomly with a fixed probability η. This kind of noise is called *uniform attribute noise*. In this case, the noise affects only the input instance, not the output label. Shackelford and Volper (1988) probed the RAN for the problem of k-DNF expressions. k-DNF is the disjunctions of terms, where each term is a conjunction of at most k-literals. Later Bshouty et al. (2003) defined a noisy distance measure for function classes, which they proved to be the best possible learning style in an attribute noise case.

They also indicated that a concept class C, is not learnable if this measure is small (compared with C and attribution noise distribution D).

Goldman and Sloan (1995) developed a uniform attribute noise model for *product random attribute noise*, in which each attribute x_i is flipped with its own probability η_i, $1 \le i \le n$. They demonstrated that if the algorithm focuses only on minimizing the disagreements, this type of noise is nearly as harmful as malicious noise. They also proved that no algorithm can exist if the noise rate η_i ($1 \le i \le n$) is unknown and the noise rate is higher than 2ε (ε is the accuracy parameter in the PAC model). Decatur and Gennaro (1995) further proved that if each noise probability η_i (or an upper bound) is known, then a PAC algorithm may exist for the simple classification problem.

2.5.3 Learning from Statistical Query

The Statistical Query (SQ) model introduced by Kearns (1993) provides a general framework for an efficient PAC learning algorithm in the presence of classification noise. Kearns proved that if any function class can be learned efficiently by the SQ model, then it is also learnable in the PAC model, and those algorithms are called SQ-typed. In the SQ model, the learning algorithm sends predicates (x, α) to the SQ oracle and asks for the probabilities P_x that the predicate is correct. Instead of answering the exact probabilities, the oracle gives only probabilities $\hat{P}_x$ within the allowed approximation error α, which here indicates a tolerance for error, i.e., $P_x - \alpha \le \hat{P}_x \le P_x + \alpha$.

The approach that the SQ model suggested to generate noise-tolerant algorithms is successful. A large number of noise-tolerant algorithms are formulated as SQ algorithms. Aslam and Decatur (1993) presented a general method to boost the accuracy of the weak

SQ learning algorithm. A later study by Blum et al. (1994) proved that a concept class can be weakly learned with at least $\Omega\left(d^{1/3}\right)$ queries, and the upper bound for the number of queries is $O(d)$. The SQ-dimension d is defined as the number of "almost uncorrelated" concepts in the concept class. Jackson (2003) further improved the lower bound to $\Omega\left(2^n\right)$ while learning the class of parity functions in an n-bit input space.

However, the SQ model has its limitations. Blumer et al. (1989) proved that there exists a class that cannot be efficiently learned by SQ, but is actually efficiently learnable. Kearns (1993) showed that the SQ model cannot generate efficient algorithms for parity functions which can be learned in a noiseless data PAC model. Jackson (2003) later showed that noise-tolerant PAC algorithms developed from using the SQ model cannot guarantee to be optimally efficient.

2.6 Learning with Queries

Angluin (1988) initiated the area of Query learning. In the basic framework, the learner needs to identify an unknown concept f from some finite or countable concept space C of subsets of a universal set. The Learner is allowed to ask specific queries about the unknown concept f to an oracle which responds according to the queries' types. Angluin studied different kinds of queries, such as membership query, equivalence query, subset, and so forth. Different from a PAC model which requires only an approximation to the target concept, query learning is a non-statistical framework and the Learner must identify the target concept exactly. An efficient algorithm and lower bounds are described in Angluin's research. Any efficient algorithm using equivalence queries in query learning can also be converted to satisfy the PAC criterion $\Pr(error(h) \geq \varepsilon) \leq \delta$.

CHAPTER 3
DATABASE SECURITY-CONTROL METHODS

In this chapter, we will survey important concepts and techniques in the area of database security, such as compromise of a database, inference, disclosure risk, and disclosure control methods among other issues. According to the way that confidential data are released, we categorize the review of database security methods into three parts: microdata, tabular data, and sequential queries to databases. Our main efforts will concentrate on the security control of a special type of database – the statistical database (SDB), which accepts only limited types of queries sent by users. Basic SDB protection techniques in the literature are reviewed.

3.1 A Survey of Database Security

For many decades, computerized databases designed to store, manage, and retrieve information, have been implemented successfully and widely in many areas, such as businesses, government, research, and health care organizations. Statistical organizations intend to provide database users with the maximum amount of information with the least disclosure risk of sensitive and confidential data. With the rapid expansion of the Internet, both the general public and the research community have been much more attentive to the issues of the database security. In the following sections, we introduce basic concepts and techniques commonly applied in a general database.

3.1.1 Introduction

A database consists of multiple tables. Each table is constructed with rows and columns representing entities (or records) and attributes (fields), respectively. Some attributes may store confidential information such as income, medical history, financial status, etc. Necessary security methods have been designed and applied to protect the privacy of specific data from outsiders or illegal users.

Database security has its own terminology for research purposes. Therefore, first we would like to clarify certain important definitions and concepts. Those are repeatedly used in this research paper and may have varied implications under different circumstances.

When talking about the confidentiality, privacy or security of a database, we refer to the *disclosure risk* of the confidential data. A *compromise* of the database occurs when the confidential information is disclosed to illegitimate users exactly, partially or inferentially.

Based on the amount of compromised sensitive information, the disclosure can be classified into *exact disclosure* and *partial disclosure* (Denning et al. 1979, Beck 1980). *Exact disclosure* or *exact inference* refers to the situation that illegal users can infer the exact true confidential information by sending sequential queries to the database, while in the case of *partial disclosure*, the true confidential data can be inferred only to a certain level of accuracy.

Inferential disclosure or *statistical inference* is another type of disclosure, which refers to the situation that an illegal user can infer the confidential data with a high probability by sending sequential queries to the database. And the probability exceeds the threshold of disclosure predetermined by the database administrator. This is known as an inference problem, which also falls within our research focus.

There are mainly two types of disclosures in terms of the disclosure objects: *identity disclosure* and *attribute disclosure*. *Identity disclosure* occurs if the identity of a subject is linked to any particular disseminated data record (Spruill 1983). Attribute disclosure implies the users could learn the attribute value or estimated attribute value about the record (Duncan and Lambert 1989, Lambert 1993). Currently, most of the research focuses on identity disclosure.

3.1.2 Database Security Techniques

Database security concerns the privacy of confidential data stored in a database. Two fundamental tools are applied to prevent compromising a database (Duncan and Fienberg 1999): (1) restricting access and (2) restricting data. For example, a statistical office or U.S. Census Bureau disseminating data to the public may enforce administrative policies to limit users' access to data. Normally the common method used is that the database administrator assigns IDs and passwords to different types of users to restrict the access at different security levels. For example, for a medical database, doctors could have full access to all kinds of information and researchers may only obtain the non-confidential records. This security mechanism is addressed as the restricting access. When all users have the same level of access to the database, only transformed data are usually allowed to be released for the purpose of security. This protection approach which is in the data restriction category reduces disclosure risk. However, for some public databases only access control is not feasible and sufficient enough to prevent inferential disclosure. Thus both tools are complementary and may be used together. However, we prioritize our research in the second category – the data restriction approach.

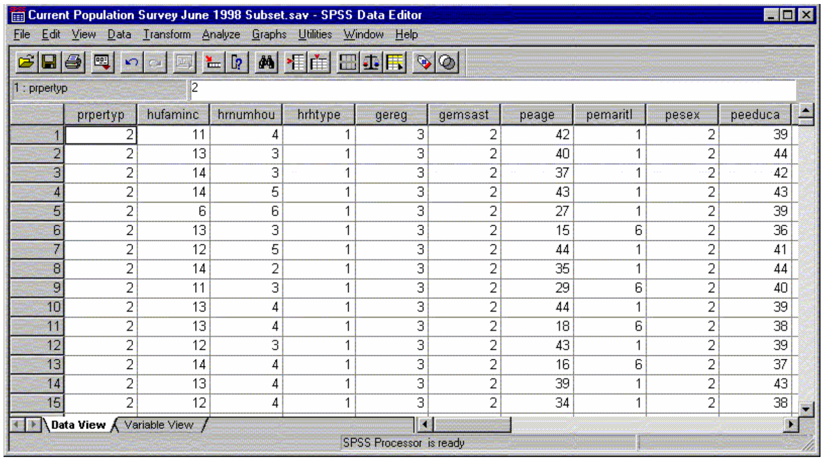
Database privacy is also known as Statistical Disclosure Control or *Statistical Disclosure Limitation (SDL).* The SDC techniques, which are used to modify original confidential data before their release, try to balance the tradeoff between information loss (or data utility) and disclosure risk. Some measures evaluating the performance of SDC methods will be discussed in Chapter 4.

Based on the way that data are released publicly, all responses from queries can be classified into three types: microdata files, tabular data files and statistical responses from sequential queries to databases (Más 2000). Most of the typical databases deal with all three dissemination formats. Our research focuses on a section of the third category – sequential queries to a statistical database (SDB), which differs from a regular database due to its limited querying interface. Normally only a few types of queries such as SUM, COUNT, Mean, and etc. can be operated in SDB.

The goal of applying disclosure control methods is to prevent users from inferring confidential data on the basis of those successive statistical queries. We briefly describe protection mechanisms for microdata and tabular data in the next two subsections, 3.1.3 and 3.1.4. Security control techniques for the statistical database are discussed in detail in section 3.2.

3.1.3 Microdata files

Microdata are unaggregated or unsummarized original sample data containing every anomynized individual record (such as person, business company, etc.) in the file. Normally, microdata originally come from the responses of census surveys issued by the statistical organizations, such as the U.S. Census Bureau (see Figure 3-1 for an example) and include detailed information with many attributes (probably over 40), such as income, occupation, household composition, and etc. Those data are released in the form of flat tables, where rows and columns represent records and attributes for each individual respondent, respectively. Microdata can usually be read, manipulated and analyzed by computers with statistical software. See Figure 3-1 for an example of microdata that are read into SPSS (Statistical Package for the Social Sciences).


Current Population Survey June 1998 Subset.sav - SPSS Data Editor
File Edit View Data Transform Analyze Graphs Utilities Window Help
Data View
Variable View
SPSS Processor is ready

1 : prpertyp 2

	prpertyp	hufaminc	hrnumhou	hrhtype	gereg	gemsast	peage	pemaritl	pesex	peeduca
1	2	11	4	1	3	2	42	1	2	39
2	2	13	3	1	3	2	40	1	2	44
3	2	14	3	1	3	2	37	1	2	42
4	2	14	5	1	3	2	43	1	2	43
5	2	6	6	1	3	2	27	1	2	39
6	2	13	3	1	3	2	15	6	2	36
7	2	12	5	1	3	2	44	1	2	41
8	2	14	2	1	3	2	35	1	2	44
9	2	11	3	1	3	2	29	6	2	40
10	2	13	4	1	3	2	44	1	2	39
11	2	13	4	1	3	2	18	6	2	38
12	2	12	3	1	3	2	43	1	2	39
13	2	14	4	1	3	2	16	6	2	37
14	2	13	4	1	3	2	39	1	2	43
15	2	12	4	1	3	2	34	1	2	38

Figure 3-1: Microdata File That Has Been Read Into SPSS.
(Data source: Indiana University Bloomington Libraries, Data Services & Resources. http://www.indiana.edu/~libgpd/data/microdata/what.html)

3.1.3.1 Protection Techniques for microdata files

Before disseminating microdata files to the public, statistical organizations will apply SDC techniques either to distort or remove certain information from original data files, therefore protecting the anonymity of individual record.

Two generic types of microdata protection methods are (Crises 2004a):

(1) Masking methods

The basic idea of masking is to add errors to the elements of a dataset before the data are released. Masking methods have two categories: perturbative (see Crises 2004d for a survey) and non-perturbative (see Crises 2004c for a survey).

The perturbative category modifies the original microdata before its release. It includes methods such as adding noise (Sullivan 1989 and Brand 2002, Domingo-Ferrer et al. 2004), rounding (Willenborg 1996 and 2000), microaggregation (Defays and Nanopoulos 1993, Anwar 1993, Mateo and Domingo 1999, Domingo and Mateo 2002, Li et al. 2002b, Hansen and Mukherjee 2003), data swapping (Dalenius and Reiss 1982, Reiss 1984, Feinberg 2000, and Fienberg and McIntyre 2004) and others.

The non-perturbative category does not change data but it makes partial suppressions or reductions of details in the microdata set, and applies methods such as sampling, suppression, recoding, and others (DeWaal and Willenborg 1995, Willenborg 1996 and 2000).

The following two tables are simple illustrations of masking methods, i.e., data swapping, Additive noise and microaggregation. (Data source: Domingo-Ferrer and Torra 2003). First the microaggregation method is used to group "Divorced" and "Widow" into one category – "Widow/er-or-divorced" in the field "Marital Status"; Secondly, values of record 3 and record 5 in the "Age" column are switched by applying data swapping techniques; finally, the value of record 4 in the "Age" attribute is perturbed from "36" to "40" by adding noise of "4".

Table 3-1: Original Records

Record	Illness	…	Sex	Marital Status	Town	Age
1	Heart	…	M	Married	Barcelona	33
2	Pregnancy	…	F	Divorced	Tarragona	40
3	Pregnancy	…	F	Married	Barcelona	36
4	Appendicitis	…	M	Single	Barcelona	36
5	Fracture	…	M	Single	Barcelona	33
6	Fracture	…	M	Widow	Barcelona	81

Table 3-2: Masked Records

Record	Illness	…	Sex	Marital status	Town	Age
1	Heart	…	M	Married	Barcelona	33
2	Pregnancy	…	F	Widow/er-or-divorced	Tarragona	40

Table 3-2. Continued.

Record	Illness	…	Sex	Marital status	Town	Age
3	Pregnancy	…	F	Married	Barcelona	33
4	Appendicitis	…	M	Single	Barcelona	40
5	Fracture	…	M	Single	Barcelona	36
6	Fracture	…	M	Widow/er-or-divorced	Barcelona	81

(2) Synthetic data generation

Liew et al. (1985) initially proposed this protection approach which first identifies the underlying density function with associated parameters for the confidential attribute, and then generates a protected dataset by randomly drawing from that estimated density function. Even though data generated from this method do not derive from original data, they preserve some statistical properties of the original distributions. However, the utility of those simulated data for the user has always been an issue. See (Crises 2004b) for an overview of this method.

3.1.4 Tabular data files

Another common way to release data is in the tabular data format (also called macrodata) obtained by aggregating microdata (Willenborg 2000). It is also called summary data, table data or compiled data. The numeric data are summarized into certain units or groups, such as geographic area, racial group, industries, age, or occupation. In terms of different processes of aggregation, published tables can be classified into several types, such as magnitude tables, frequency count tables, linked tables, etc.

3.1.4.1 Protection techniques for tabular data

Tabular data files collect data at a higher level of aggregation since they summarize individual atomic information. Therefore they provide higher security for database than microdata files. However, the disclosure risk has not been completely eliminated and intruders could still infer confidential data from an aggregated table (see Table 3-3 and

3.4 for an example). Protection techniques, such as cell suppression (Cox 1975, 1980, Malvestuto et al. 1991, Kelly et al. 1992, Chu 1997), table redesign, noise adding, rounding, or swapping among others, have to be adopted before the release. See Sullivan (1992), Willenborg (2000), Oganian (2002) for an overview.

See Table 3-3 for an illustration of tabular data. It shows state level data for various types of food stores The Economic Division published the economic data by geography and standard industrial classification (SIC) codes. The “Value of Sales” field is considered as confidential data. Table 3-4 demonstrates how a cell suppression technique is applied to protect the confidential data. (Data source: U.S. Bureau of the Census Statistical Research Division, Sullivan 1992).

Table 3-3: Original Table:

SIC		…	Number of Establishments	Value of Sales ($)
54	All Food Stores	…	347	200,900
541	Grocery	…	333	196,000
542	Meat and Fish	…	11	1,500
543	Fruit Stores	…	2	2,400
544	Candy	…	1	1,000

Table 3-4: Published Table After Applying Cell Suppression

SIC		…	Number of Establishments	Value of Sales ($)
54	All Food Stores	…	347	200,900
541	Grocery	…	333	196,000
542	Meat and Fish	…	11	1,500
543	Fruit Stores	…	2	D
544	Candy	…	1	D

Only one Candy store reported sales value for this state in Table 3-3. If the table is released as it is, any user would learn the exact sales value for this specific store. Also a sales value is listed for two Fruit stores in this state. Therefore by knowing its own sales figure, either of these two stores can infer the competitor’s sales volume. A disclosure occurs under either situation. Thus, SDC methods have to be incorporated into the original table before its publication.

Table 3-4 shows that the confidential data resulting in a compromise are suppressed and replaced by a “D” in the cells. The technique applied is called cell suppression, which is very commonly used by U.S Bureau Census currently.

3.2 Statistical Database

3.2.1 Introduction

A statistical database (SDB) differs from a regular database due to its limited querying interface. Its users can retrieve only aggregate statistics of confidential attributes, that is, SUM, COUNT, and Mean, for a subset of records stored in the database. Those aggregate statistics are calculated from tables in databases. Tables could include microdata or tabular data. In other words, query responses in SDBs could be treated as views of microdata or tabular data tables. However, those views can only be summarized to answer limited types of queries and in the form of aggregate statistics they are computed according to each query. A SDB is compromised if the sensitive data is disclosed by answering a set of queries. Note that some of the protection methods used in SDBs are overlapped with those for microdata files and tabular data files. However, SDBs security methods emphasize on preventing a disclosure from responding sequential queries.

Many government agencies, businesses, and research institutions normally collect and analyze aggregate data for their special purposes. For instance, medical researchers may need to know the total number of HIV-positive patients within a certain age range and gender. The users should not be allowed to link the sensitive information to any specific record in the SDB by asking sequential statistical queries. We illustrate how a statistical database could possibly be compromised by the following example, and further explain the necessity of applying statistical disclosure control methods before data are released.

3.2.2 An Example: The Compromise of Statistical Databases

Adam and Wortmann (1989) described three basic types of authorized users for a statistical database: the non-statistical users accessing the database, sending queries and updating data; the researchers authorized to receive only aggregate statistics; and the snoopers, attackers or adversaries seeking to compromise the database. The purpose of database security is to provide researchers with useful information while preventing disclosure risk from attackers.

For instance (example from Adam and Wortmann 1989, Garfinkel et al. 2002), a hospital's database (see Table 3-5) providing aggregate statistics to the outsiders contains one confidential field, that is, HIV status which is denoted by "1" as positive and "0" as otherwise. Suppose a snooper knows that Cooper working for company D is a male under the age of 30, and attempts to find out whether or not Cooper is HIV-positive. Therefore, he types the following queries:

Query 1: Sum = (Sex=M) & (Company=D) & (Age<30);

Query 2: Sum = (Sex=M) & (Company=D) & (HIV=1) & (Age<30);

The response to Query 1 is 1, and the response to Query 2 is 1.

Neither of queries is a threat to the database privacy individually, however, when they are put together, the attacker who knows Cooper's personal information can locate Cooper from Query 1's answer and immediately infer that Cooper is HIV-positive from Query 2's answer. Thus, the confidential data is disclosed. And we refer to this case as a compromise of a database.

From this example, we can tell that the snooper is able to infer the true confidential data through analyzing aggregate statistics by sending the sequential queries. Therefore security mechanisms have to be established prior to the data release.

Table 3-5: A Hospital's Database (data source: part from Garfinkel et al. 2002)

Record	Name	Job	Age	Sex	Company	HIV
1	Daniel	Manager	27	F	A	0
2	Smith	Trainee	42	M	B	0
3	Jane	Manager	63	F	C	0
4	Mary	Trainee	28	F	B	1
5	Selkirk	Manager	57	M	A	0
6	Daphne	Manager	55	F	B	0
7	Cooper	Trainee	21	M	D	1
8	Nevins	Trainee	32	M	C	1
9	Granville	Manager	46	M	C	0
10	Remminger	Trainee	36	M	D	1
11	Larson	Manager	47	M	B	1
12	Barbara	Trainee	38	F	D	0
13	Early	Manager	64	M	A	1
14	Hodge	Manager	35	M	B	0

3.2.3 Disclosure Control Methods for Statistical Databases

Some basic security control methods for microdata and tabular data have been summarized in the previous sections. In this section, we will concentrate on the security control methods for statistical databases. Some methods used for microdata and tabular data may also be utilized here. Adam and Wortmann (1989) conducted a complete survey about security techniques for statistical databases (SDBs). They classified all security methods for SDBs into four categories: conceptual, query restriction, data perturbation, and output perturbation. In addition to that, Adam and Wortmann provided five criteria to evaluate the performance of security mechanisms. Our literature review will follow suit and discuss major security control methods in the following sections.

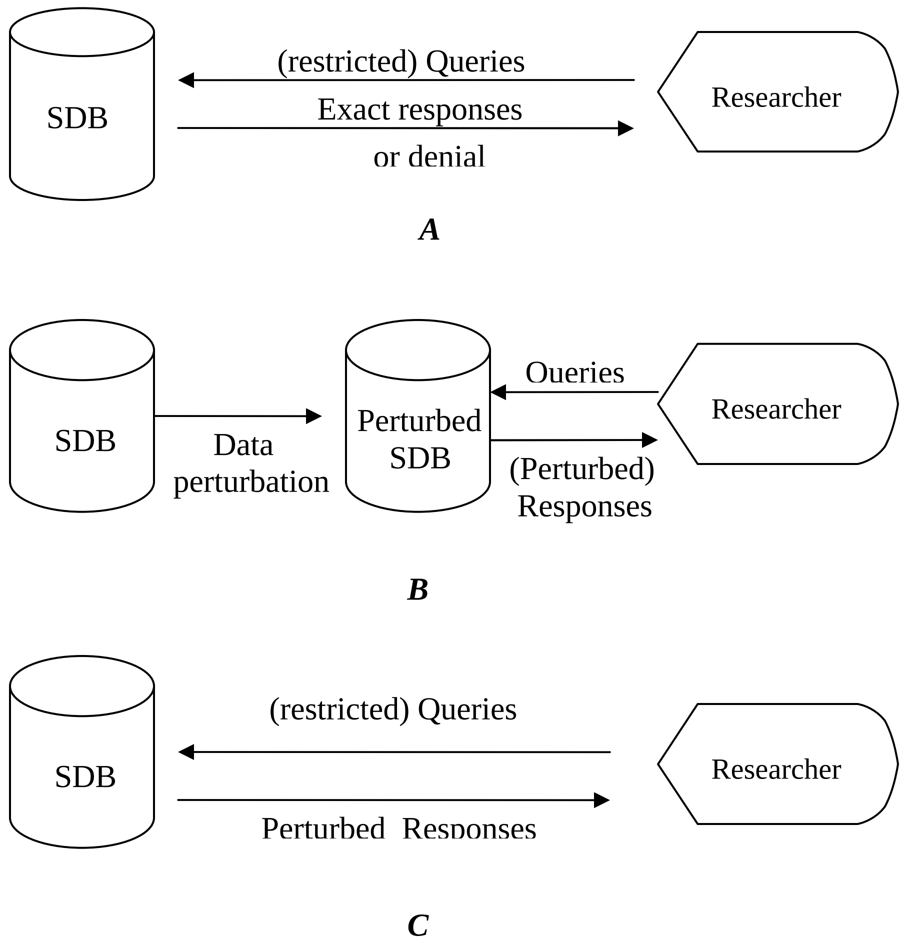


Figure 3-2: Three Approaches in Statistical Database Security. A) Query Restriction, B) Data Perturbation and C) Perturbed Responses.

Figure 3-2 demonstrates three approaches: Query Restriction, Data Perturbation and Output Perturbation (Data source: Adam and Wortmann 1989). Figure 3-2A shows how Query Restriction method works. This technique either returns exact answers to the user or refuses to respond at all. Figure 3-2B introduces Data Perturbation method which creates a perturbed SDB from the original SDB to respond to all queries. The user can receive only perturbed responses. The output perturbation method is illustrated in Figure 3-2C. Each query answer is modified before being sent back to the user.

3.2.3.1 Conceptual approach

The Conceptual approach includes two basic models: the Conceptual and Lattice models. The Conceptual model, proposed by Chin and Ozsoyoglu (1981, 1982), addressed security issues at a Conceptual data model level where the users only access entities with common attributes and their statistics. The Lattice model developed by Denning (1983) and Denning and Schlorer (1983), retrieved data from SDBs in tabular form at different aggregation levels. Both methods provide a fundamental framework to understand and analyze SDBs' security problems, but neither seems functional at the implementation level.

3.2.3.2 Query restriction approach

Based on the users' query history, SDBs either provide the exact answer or decline the query (see Figure 3-2A). The five major methods in this approach include:

(1) Query-set-size control (Hoffman and Miller 1970, Fellegi 1972, Schlorer 1975 and 1980, Denning et al. 1979, Schwartz et al. 1979, Denning and Schlorer 1980, Friedman and Hoffman, 1980, Jonge 1983). This method allows the release of the data only if the query set size (number of records included in the query response) meets some specific conditions.

(2) Query-set-overlap control (Dobkin et al. 1979). This mechanism is based on query-set-size control and further explores the possible overlapped entities involved in successive queries.

(3) Auditing (Schlorer 1976, Hoffman 1977, Chin and Ozsoyoglu 1982, Chin et al. 1984, Brankovic et al. 1997, Malvestuto and Moscarini 1998, Kleinberg et al. 2000, Li et al. 2002a, Malvestuto and Mezzini 2003). This technique intends to keep query records

for each user, and before answering new queries, it checks whether or not the response can lead to a disclosure of the confidential data.

(4) Partitioning (Yu and Chin 1977, Chin and Ozsoyoglu 1979, 1981, Schlorer 1983). This method groups all entities into a number of disjoint subsets. Queries are answered on the basis of those subsets instead of original data.

(5) Cell suppression (Cox 1975, 1980, Denning et al. 1982, Sande 1983, Malvestuto and Moscarini 1990, Kelly et al. 1992, Malvestuto 1993). The basic idea of the technique is to suppress all cells that may result in the compromise of SDBs.

So far, some methods in this category have been proved either inefficient or infeasible. For instance, a statistical database normally includes a large number of data records. Under this situation, a traditional auditing method would become impractical due to its requirement for large memory storage and strong computing power. Among those methods, the most promising method is the cell suppression technique, which has been implemented successfully by the US Census Bureau and widely adopted in the real world.

3.2.3.3 Data Perturbation Approach

In this approach, a dedicated perturbed database is constructed once and for all by altering the original database to answer users' queries (see Figure 3-2B). According to Adam and Wortmann (1989), all methods fall into two categories:

(1) The probability distribution. This category treats SDB as a sample drawn from some distribution. The original SDB is replaced either by another sample coming from the same distribution, or by the distribution itself (Lefons et al. 1983). Techniques in this category include data swapping (Reiss 1984), multidimensional transformation of

attributes (Schlorer 1981), data distortion by probability distribution (Liew et al. 1985), and etc.

(2) Fixed data perturbation. This category includes some of the most successful database protection mechanisms. It can be achieved by either an additive or multiplicative technique (Muralidhar et al. 1999, 1995). An additive technique (Muralidhar et al. 1999) refers to adding noise to the confidential data. The multiplicative data perturbation (Muralidhar et al. 1995) protects the sensitive information by multiplying the original data with a random variable, which has mean of 1 and a prespecified variance. Our study focuses on the additive data perturbation, which are classified into two types of perturbation in our research: random data perturbation and variable data perturbation. We will introduce these two methods separately in Chapter 5.

3.2.3.4 Output Perturbation Approach

Output Perturbation is also named query-based perturbation. The response for each query is computed first from the original database, and then it is perturbed based on the answer of each query (see Figure 3-2C). Three methods are included in this approach:

(1) The Random-Sample Queries technique is proposed by Denning (1980). Later, Leiss (1982) suggested a variant of Denning’s method. The basic rationale is that the query response is calculated from a randomly selected sampled query set. This selected query set is chosen from the original query set by satisfying some specific conditions. However, an attacker may compromise the confidential information by repeating the same query and averaging the results.

(2) Varying-Output Perturbation (Beck 1980) works for SUM, COUNT and Percentile queries. This method assigns a varying perturbation to the data that are used to compute the response statistic.

(3) Rounding includes three types of output perturbation: systematic rounding (Achugbue and Chin 1979), random rounding (Fellegi and Phillips 1974, Haq 1975, 1977), and controlled rounding (Dalenius 1981). This technique calculates queries based on unbiased data, and then the answer is rounded up or down to the nearest multiple of a base number set by Database Administrators (DBAs). Query results do not change for the same query, therefore providing good protection in terms of averaging attacks.

In this chapter we summarized different types of database security-control methods. For a specific database, one SDC method could be more effective and efficient than another. Therefore, how to select the most suitable security method becomes a critical issue in the database privacy. We will review various performance measurements for SDC in the next chapter.

CHAPTER 4
INFORMATION LOSS AND DISCLOSURE RISK

Chapter 2 provided an overview of important SDC methods that are applied to protect the privacy of a database. However, since SDC methods reach their goals by transforming original data, users of the database would achieve only approximate results from a modified data. Therefore, a fundamental issue that every statistical organization has to address is how to protect confidential data maximally while providing database users with as much useful and accurate information as possible. In this chapter, we review the main performance measurements of SDC methods. These assessments are used to evaluate the information loss (used interchangeably with data utility) and disclosure risk of a database. These measures have become standard criteria for deciding on how to choose appropriate protection techniques for SDBs.

4.1 Introduction

All SDC methods attempt to optimize two conflicting goals:

(1) Maximizing data utility or minimizing information loss that legitimate data users can obtain.

(2) Minimizing the disclosure risk of the confidential information that data organizations take by publishing the data.

Therefore the efforts to obtain greater protection usually result in reducing the quality of data that are released. So the database administrators always seek to solve the problem by optimizing tradeoffs between the information loss and disclosure risk. The definitions for information loss and disclosure risk are as follows:

Information Loss (IL) refers to the loss of the utility of data after being released. It measures the damage of the data quality for the legal users due to the application of SDC methods.

Disclosure Risk (DR) refers to the risk of disclosure of confidential information in the database. It measures how dangerous it is for statistical organizations to publish modified data.

The problem that statistical organizations always have to confront is how to choose an appropriate SDC method with suitable parameters from many potential protection mechanisms. And the selected mechanism should be able to minimize disclosure risk as well as information loss. One of the best solutions is to count on performance measures to evaluate the suitability of different SDC techniques to the database. Good designs for performance criteria quantifying information loss and disclosure risk are therefore desirable and necessary.

4.2 Literature Review

Designing good performance measures is a challenging task because different users collect data for different purposes and organizations define disclosure risk to different extents. So far, there are many performance assessment methods existing in the literature. Based on their properties, we divide those measurement techniques into five categories in our research:

(1) Information loss measures for some specific protection methods.

This type of measurement assesses the difference of masked (modified) data from original data after applying a specific protection method. Refer to Willenborg and Waal (2000) and Oganian (2002) for example. If variances of the original microdata are critical for the user, then the information loss can be estimated as

$$Var\left(\hat{\theta}\left(data_{masked}\right)\right)\Big/Var\left(\hat{\theta}\left(data_{original}\right)\right)$$

where $\hat{\theta}\left(data_{original}\right)$ is a consistent estimator of the original data, and $\hat{\theta}\left(data_{masked}\right)$ is the corresponding estimator of the modified data. We can tell from the above criterion that this measurement depends on a specific purpose of data use, such as mean, variances, etc.

(2) Generic information loss measures for different protection methods.

A generic information loss measure, which is not limited to any particular data use, is designed to compare different protection methods. Two well-known general information loss measures are as follows:

Shannon's entropy, discussed in Kooiman et al. (1998) and Willenborg and Waal (2000), can be applied to any SDC technique to define and quantify information loss. This measurement models the masking process as noise added to the original dataset, which then is sent through a noisy channel. The receiver of the noisy data intends to reconstruct the probability distribution of the original data. The entropy of this probability distribution measures the uncertainty of the original data after masked data are released because of the transmission process. However an entropy-based measurement is not a very good criterion since it ignores the impact of covariances and means. Whether or not these two statistics can be preserved properly from the original data directly affects the validity and quality of the altered data.

Another measurement by Domingo-Ferrer et al. (2001) and Oganian (2002) suggests that IL would be small if the original and masked data have similar analytical structure, but the disclosure risk would be higher in this case. This method compares statistics, such as mean square error, mean absolute error, and mean variation, which are calculated from the difference of covariance matrix, coefficient matrix, correlation matrix, and etc. between the original data and modified data.

(3) Disclosure risk measures for specific protection methods.

The disclosure risk also affects the quality of the SDC methods. Compared with IL measures, DR measures are more method-specific. The idea of assessing disclosure risk was initially proposed by Lambert (1993). Later, different DR measures were developed for SDC methods, i.e., for sampling methods by Chen and Keller-McNulty (1998), Samuel (1998), Skinner et al. (1994), and Truta et al. (2004), and for micro-aggregation masking methods by Jaro (1989), and Pagliuca and Seri (1998).

(4) Generic disclosure risk measures for different protection methods.

The two main types of general DR measurements are applied to measure the quality of different protection methods for tabular data. The first measurement is called sensitivity rules, which is used to estimate DR prior to the publication of data tables. There are three methods: (n,k)-dominance, $p\%$-rule, and pq rule (Felso et al. 2001, Holvast 1999, Luige and Meliskova 1999). Different from dominance rule, which is criticized for its failure to to reflect the disclosure risk properly, a new priori measure is proposed by Oganian (2002), who also introduced a posterior DR measure, which takes the modified data into account and operates after applying SDC methods.

A new method based on Canonical Correlation Analysis was introduced by Sarathy and Muralidhar (2002) to evaluate the security level for different SDC methods. This methodology can also be used to select the appropriate inference control method. For more details, refer to Sarathy and Muralidhar (2002).

(5) Generic performance measures that encompass disclosure risk and information loss for different protection methods.

A sound SDC method should be able to achieve an optimal tradeoff between disclosure risk and information loss. Therefore a joint framework is desired to examine the tradeoffs and compare the performance of distinct SDC methods. Two popular performance measures in the literature are Score Construction and R-U confidentiality map.

Score Construction, proposed by Domingo-Ferrer and Torra (2001), ranks different SDC methods, based on their scores obtained by averaging their information loss and disclosure risk measures. For example (Crisis 2004e),

$$Score(V,V^{'}) = \frac{IL(V,V^{'}) + DR(V,V^{'})}{2}$$

Where V is the original data, $V^{'}$ is the modified data. Information Loss (IL) and Disclosure Risk (DR) are information loss and disclosure risk measures. Refer to Crisis (2004e), Domingo-Ferrer et al. (2001), Sebé et al. (2002) and Yancey et al. (2002) for more examples.

An R-U confidentiality map, first proposed by Duncan and Fienberg (1999), constructs a general analytical framework for information organization to trace the tradeoffs between disclosure risk and data utility. It was further developed by Duncan et al. (2001, 2004), and Gomatam et al. (2004). Trottini and Fienberg (2002) later illustrated two examples of R-U map in their paper. An application is given in Boyen et al. (2004). Database administrators could decide the most appropriate SDC method from the R-U map by observing the influence of a particular method with the according parameter choice. See the following figure (Data source: Trottini and Fienberg 2002) for an example.

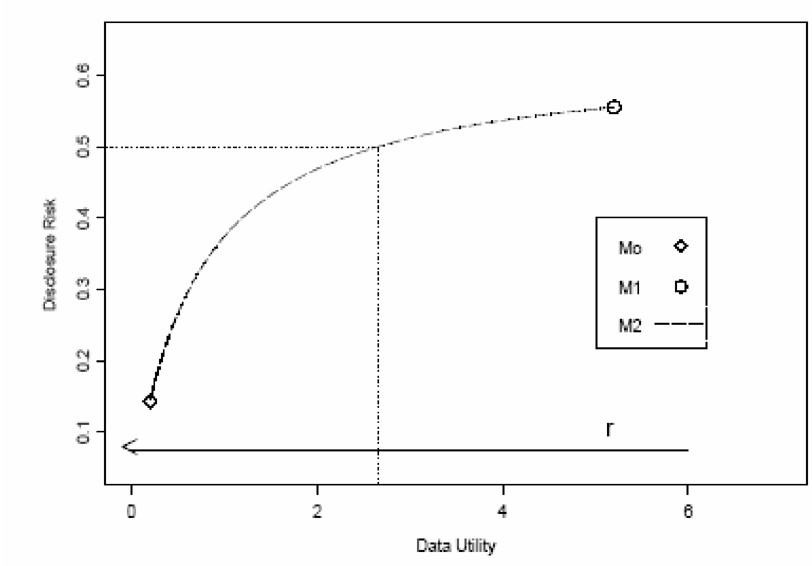


Figure 4-1: R-U Confidentiality Map, Univariate Case, $n=10$, $\phi^2=5$, $\sigma^2=2$

M_0, M_1 *and* M_2, are represented by a diamond, a circle and a dashed line in the figure, and indicate three types of SDC methods: trivial microaggregation, microaggregation, and the combination of additive noise and microaggregation, respectively. The disclosure risk and data utility are functions determined by the data size n, known variance (prior belief) ϕ^2, known population variance σ^2, and the standard deviation r of the noise added to the original data. The y-axis measures the disclosure risk while the x-axis estimates the data utility. For example, checking Figure 3-2, if the database administrators intend to have the disclosure risk below 0.5, we will see that the appropriate SDC method that satisfies this requirement is M_2, the mixed strategy of additive noise plus microaggregation method. From the x-axis, the corresponding data utility is shown as 2.65. The choice of r can also affect the R-U map. If r is large, then the mixed strategy M_2 is close to not release any data at all, as r is chosen close to zero, the M_2 is equivalent to the microaggregation method with some specific parameter. In Figure 4-1, $r = 2.081$.

We do not differentiate the measurements for microdata and tabular data in the overview since our research focuses on statistical databases. All examples and methods previously mentioned are applied either to microdata or tabular data or both.

CHAPTER 5
DATA PERTURBATION

This chapter provides an introduction to additive data perturbation methods. Based on different ways of generating perturbative values, additive data perturbation methods are classified into three categories: random-data perturbation, fix-data perturbation and variable-data perturbation. The first category, random-data perturbation, with five types of perturbation methods, can be found in Kim 1986, Muralidhar et al. 1999, Sullivan 1989, Tendick 1991, Tendick and Matloff 1994. Our proposed variable-data perturbation method is a new category that includes the interval protection technique given by Gopal et al. (1998, 2002) and Garfinkel et al. (2002). In both random data perturbation and variable-data perturbation methods, a perturbed database is constructed by adding noise to the confidential data in the original database. All query responses are computed from the perturbed database. We will review an algorithm by Dinur and Nissim (2003) that finds a bound for the fixed-data perturbation. The noise is added to each query response. This bound can be applied to both data perturbation and output perturbation methods. Their work considers the tradeoff between privacy and usability of a statistical database. We end the chapter with the proposed approach to the database security problem.

5.1 Introduction

Our study focuses on additive noise perturbation methods, which are usually employed to protect confidential numerical data. Perturbation methods can guarantee the prevention of the exact disclosure by adding noise to sensitive data, however they are still susceptible to partial disclosure and inferential disclosure. (See Chapter 3 for definitions of exact disclosure, partial disclosure and inferential disclosure.)

Two types of additive perturbation methods are described in the following sections based on their different approaches of generating noise. An algorithm by Dinur and Nissim (2003) providing a theoretical basis for our study is also reviewed. Our proposed research approach is discussed at the end of this chapter.

5.2 Random Data Perturbation

5.2.1 Introduction

Random Data Perturbation (RDP) is one of the most popular and practical data protection methods employed in statistical databases today. In order to effectively prevent statistical inference against a snooper, DBAs attempt to provide an appropriate level of security by distorting the sensitive data with random noise. The RDP method could assure adequate protection of confidential information while satisfying legitimate users' needs for aggregate statistics of the database.

5.2.2 Literature Review

In the Random Data Perturbation (RDP) method, a perturbed database is created by adding random noise to the confidential numerical attribute(s). We discuss four types of RDP summarized by Crises (2004) and describe a general method for RDP given by Muralidhar et al. (1999).

Before walking through different types of RDP methods, we first discuss the main disadvantage of the data perturbation methods. RDP methods may generate bias into statistical characteristics of databases, such as PERCENTILES, conditional SUMS, and COUNTS. Matloff (1986) initially introduced the concept of *bias*, which occurs when the responses to certain queries computed from a perturbed database may be different from the responses computed from the original database. The four types of bias, A, B, C, and D, are defined and analyzed in the literature by Muralidhar et al. (1999). Type A bias occurs when a change in variance causes a change of summary measures of some perturbed attribute. Typed B bias applies when the perturbation distort the relationships between confidential attributes. Type C bias occurs when the perturbation changes the relationships between confidential and non-confidential attributes. Type D bias occurs when the underlying distribution of the perturbed database can not be determined because the original database or noise term has a non-multivariate normal distribution. Improved perturbation methods are designed to avoid bias (Matloff 1986, Tendick 1991, Tendick and Matloff 1994, Muralidhar et al. 1995). A creative method called General Additive Data Perturbation (GADP), proposed by Muralidhar (1999), deletes all these types of bias completely from additive perturbation methods. For more information about GADP, see Section 5.2.

(1) Masking by uncorrelated noise addition

This method is also called the Simple Additive Data Perturbation method (Muralidhar et al. 1999). The vector of confidential fields, d_m, representing the m^{th} attribute of the original database which contains n records, is replaced by a vector y_m by adding a noise term e_m:

$$y_m = d_m + e_m$$

where each element of e_m is normally distributed and drawn from a random variable $\pi_m \sim N\left(0,\ \sigma^2_{\pi_m}\right)$. Each noise term is generated independently of the others, such that $Cov\left(\pi_i, \pi_j\right) = 0$ for all $i \neq j$. The variances of π_m are generally assumed proportional to those of the original vector d_m, that is, if the variance of d_m is σ_m^2, then $\sigma_{\pi_m}^2 := \alpha\sigma_m^2$. The distribution of π_m and parameter α are decided by the DBA. This perturbation method introduces Type A, B and C bias.

(2) Masking by correlated noise addition

This method proposed by Kim (1986) and Tendick (1991) uses correlated noise to perturb the database. It is also called the Correlated-Noise Additive Data Perturbation method (CADP). The formulation of the method is:

$$V_y = V + V_\pi$$

where V_y is the covariance matrix from the perturbed data; V_π is the covariance matrix of the errors, that is, $\pi \sim N\left(0,\ V_\pi\right)$, which is proportional to the covariance matrix of the original data, V, that is:

$$V_\pi = \alpha V$$

The CADP method generates Type A and Type C bias.

(3) Masking by noise addition and linear transformations

In Kim (1986), Tendick and Matloff (1994), Crises (2004), and Muralidhar et al. (1999) masking by correlated noise addition was modified to use additional linear transformations to eliminate certain types of bias. Therefore, the sample covariance matrix of the masked data is an unbiased estimator for the covariance matrix of the original data. This method is also named the Bias-Corrected Correlated-Noise Additive Data Perturbation (BCADP) method and only results in Type C bias.

(4) Masking by noise addition and nonlinear transformation

Sullivan (1989) proposed a complex algorithm (not discussed here) combining simple additive noise with a nonlinear transformation. This masking method is applied to discrete attributes.

Muralidhar et al. (1999) introduced a General Method for the Additive Data Perturbation (GADP) method, which is a further improvement on the previous RDP methods. Suppose the database U has a set C of confidential attributes and a set NC of non-confidential attributes with n records. A perturbed database P which only alters the attributes in set C is constructed on the basis of the original database U. The perturbation process keeps all statistical relationships, such as the mean values for C, and measures of the covariance and canonical correlation between C and NC. Then each record in the set C is generated from a multivariate normal distribution. This process is repeated for all records. The GADP method guarantees that the statistical properties between all attributes are the same before and after perturbation, therefore eliminating all types of bias. Thus, the GADP is called a bias-free RDP method. By comparing with other perturbation methods empirically, Muralidhar et al. suggested that the GADP method would provide the highest level of security and represents a general form of additive noise perturbation.

5.3 Variable Data Perturbation

5.3.1 CVC Interval Protection for Confidential Data

Gopal, Goes, and Garfinkel (1998) initiated the idea of interval protection for confidential information in a database and introduced the concept of interval disclosure. They developed three techniques, which they called “Technique-LP”, “Technique-ELS, and Technique-RP”, for various query types. As a result, the query types that a user could ask are limited to SUM (COUNT), Mean, MIN, and MAX for numerical data. This method was further studied in Gopal et al. (2000). Later, Gopal et al. (2002) formally proposed the Confidentiality via Camouflage (CVC) interval protection technique, which is designed to answer numerical ad hoc statistical queries to an online database. Garfinkel et al. (2002, 2004) further extended this technique.

Garfinkel et al. (2002) explored the CVC technique for privacy protection of binary confidential data and answered only ad hoc COUNT queries (the same as SUM queries here). The extended technique is called Bin-CVC. Consider a database consisting of n records. The Bin-CVC technique introduces s binary camouflage vectors, $P = \left\{ P^1, P^2 ..., P^{s-1}, P^s \right\}$, which are used to camouflage or hide the true confidential vector d, where $P^s = d$ for s. Without loss of generality, they assumed the database contained only one binary confidential field. Each camouflage vector is denoted as $P^j = \left(p_1^j, ..., p_n^j \right)$. When a user asks a query q, an interval answer $I(q) = \left[l(q),\ u(q) \right]$ will be returned as follows. The upper bound $u(q)$ and lower bound $l(q)$ of the interval are calculated from the maximum and minimum of all camouflage vectors in the specific set related to the query, that is, $u(q) = \max_{j \in P} \sum_{i \in q} p_i^j$ and $l(q) = \min_{j \in P} \sum_{i \in q} p_i^j$. The true answers are guaranteed to be inside the interval response, $\sum_{i \in q} d_i \in I(q)$.

Table 5-1: An Example Database (Data source: Garfinkel et al. 2002)

Record	Name	Job	Age	Company	HIV
1	Jones	Manager	27	A	0
2	Smith	Trainee	42	B	0
3	Johnson	Manager	63	C	0
4	Andres	Trainee	28	B	1
5	Selkirk	Manager	57	A	0
6	Clark	Manager	55	B	0
7	Cooper	Trainee	21	D	1
8	Nevins	Trainee	32	C	1

Table 5-1. Continued

Record	**Name**	**Job**	**Age**	**Company**	**HIV**
9	Granville	Manager	46	C	0
10	Brady	Trainee	36	D	1
11	Larson	Manager	47	B	1
12	Remminger	Trainee	28	D	0
13	Early	Manager	64	A	1
14	Hodge	Manager	35	B	0

The HIV status field represents a binary confidential field with 14 records (see Table 5-1). All query responses involving this sensitive field are computed from camouflage vectors generated by the Bin-CVC technique. Table 5-2 is an example of camouflage vectors for this database where vector P^3 is the true vector.

Table 5-2: The Example Database with Camouflage Vector(Data source: Garfinkel et al. 2002)

Record	**P^1**	**P^2**	**P^3 = d**
1	1	0	0
2	0	1	0
3	1	0	0
4	0	0	1
5	0	1	0
6	1	0	0
7	0	0	1
8	0	0	1
9	0	1	0
10	0	0	1
11	0	0	1
12	1	0	0
13	0	0	1
14	0	1	0

Camouflage vectors are generated from a complex network algorithm. The design of the network algorithm whose joint paths construct different camouflage vectors is a critical step in the success of the Bin-CVC model. The network represents all n records in the confidential field with variables $\left(x_1,\cdots,x_n\right)$. All paths start from the source to the destination. The network is constructed using two parameters. Parameter w gives the total number of paths, and parameter m is the number of paths consisting only of true value edges. These determine the number of camouflage vectors $s = \binom{w}{m}$. An illustration of the network construction of the example database (see Table 5-1) using three camouflage vectors (see Table 5-2) is shown in Figure 5-1.

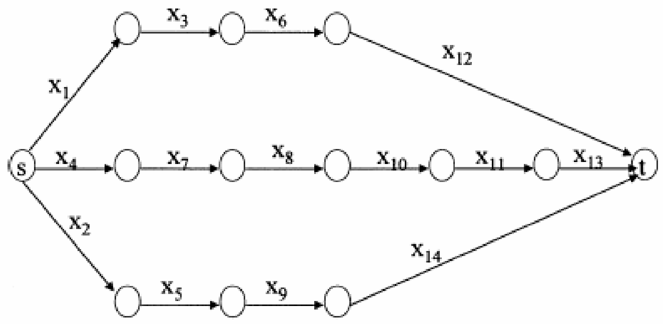


Figure 5-1: Network With $(m, w) = (1, 3)$ (data source: Garfinkel et al. 2002)

In the example database (Table 5-1), all 14 records in the confidential field are denoted by variables $(x_1, \cdots, x_{14})$. Parameter $w = 3$ indicates 3 disjoint paths are constructed in the network and $m = 1$ implies that all those variables with true value 1 in the true confidential field are assigned to one of three paths. Variables representing other records with value zero are assigned as evenly as possible to the rest of two paths. The total number of camouflage vectors is $s = \binom{3}{1} = 3$. Every camouflage vector is the combination of choosing m edges out of w paths. So, in Figure 5-1, each camouflage vector selects one edge out of three paths with their true value records on the path. Compared with Table 5-2, camouflage vector P^1 has records 1, 3, 6, and 12 containing value one. The remaining records in P^1 are zero. In the corresponding network, accordingly there is one path including only variables (x_1, x_3, x_6, x_{12}).

Performance measurement $CB = 1 - \left| p* - m/w \right|$ is employed to assess the quality of networks for a given database with different w and m values, where CB stands for Column Balancing. The usefulness of each query answer is computed by the formula: $Z = 100 \times \left(1 - \left(u(q) - l(q)\right)/\left|q\right|\right)$. $\left|q\right|$ denotes the cardinality of the query q which is the number of records that are involved in that query. The closer to 1.0 Z is, the better the query answer is.

The ideal network that yields the tightest interval response has a small s and every camouflage vector has the same number of ones as the true confidential field. That is, $p^j = p*$, where p^j is the proportion of ones in P^j, and $p*$ is the proportion of ones in $P^s = d$. This ideal structure is called “perfect column balancing”. See Table 5-2 as an example. Here $p^1 = p^2 = 0.4$, $p* = 0.6$. A good CB “increases the probability of (a) better query answer”.

Bin-CVC is a very promising methodology for the database privacy. However, instead of an exact answer, it responds to the query with an interval which reduces the data utility. We define the information loss of the CVC technique as the width of the interval, given by $e_q = u(q) - l(q)$.

5.3.2 Variable-data Perturbation

Inspired by the CVC technique, we propose a new data perturbation method - the variable-data perturbation. Different from random data perturbation whose random noise is drawn from a normal distribution $N\left(0, \sigma^2\right)$, the *variable-data perturbation* method is defined as a data perturbation method which modifies the confidential information by adding discrete noise that is generated by a parametrically driven algorithm, such as w

and m in the CVC interval protection method. The perturbed database is created once and for all. The algorithm can choose various parameters to produce different types of noise. We can view the output of the algorithm as if it were pulling values randomly from some distribution D with known parameters, with a non-zero mean μ and variance σ^2. The mean and variance are always finite. Each query answer is computed from the perturbed data.

A discrete random data perturbation method builds a perturbed database from which all query responses are computed. Output perturbation method does not alter the database, but query answers are perturbed before they are returned to the user. Variable data perturbation method is a hybrid of data perturbation and output perturbation and generates noise for the confidential field. Perturbed answers for each query involving sensitive data are calculated only from the perturbed confidential vector. We treat the variable-data perturbation as a data perturbation method with query protection.

Consider the Bin-CVC technique as an example of the variable-data perturbation method. The network algorithm creates camouflage vectors to disguise the true confidential vector once and for all. Each query answer is an interval which is computed from the camouflage vectors and assures the true answer is included. In a worst-case scenario, the noise or perturbation could be regarded as the difference between the lower bound and upper bound of the interval: $e_q = u(q) - l(q)$, where e_q are discrete random variable.

We simulated the network algorithm on the example database (see Table 5-1) in Garfinkel et al. (2002) and computed the interval answers for all queries. Since the confidential vector in the database is a 14-bit binary string, the total number of queries involving this binary vector is 2^{14}. The following figures (Figure 5-2 A-D) show four different cases with parameters of the network algorithm at (1) $w = 5$ and $m = 2$; (2) $w = 7$ and $m = 3$; (3) $w = 8$ and $m = 5$; (4) $w = 12$ and $m = 6$. Among those networks, $w = 7$ and $m = 3$ creates perfect column balancing and based on its frequencies of each noise value for all 2^{14} queries, we obtain a noise distribution with mean $\mu = 3.302$ and variance $\sigma^2 = 1.379$ as shown in Figure 5-2B.

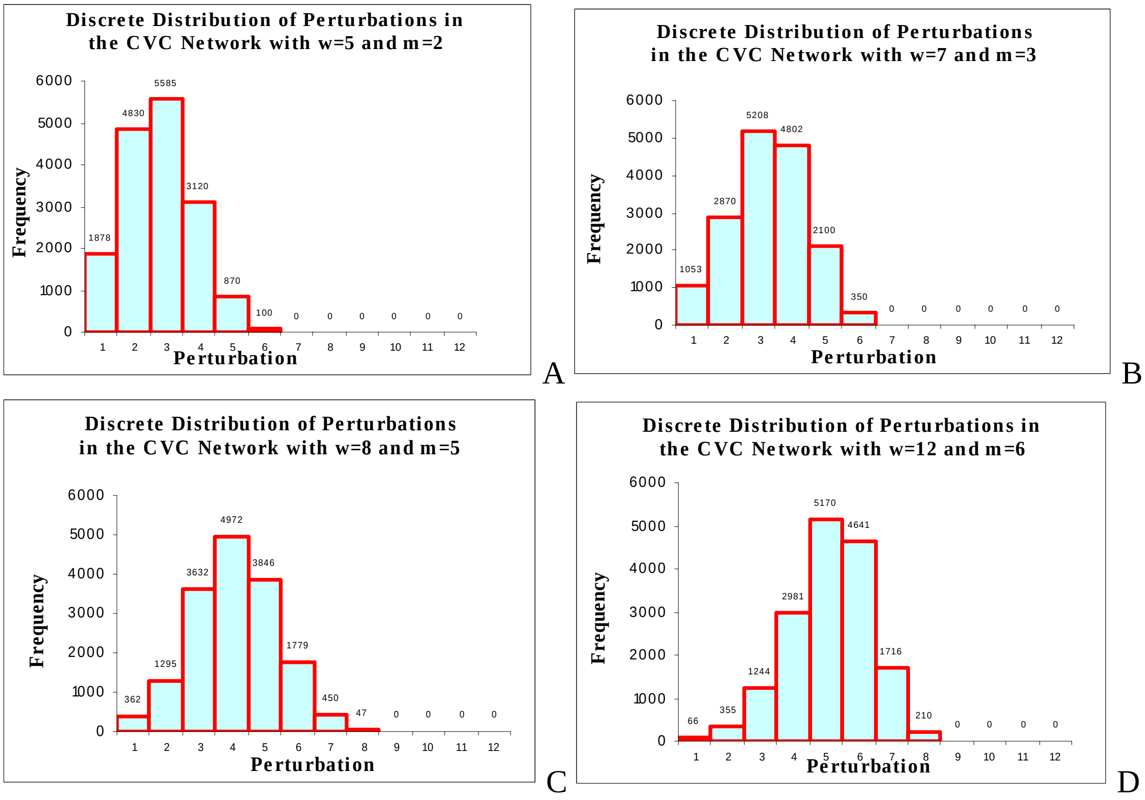


Figure 5-2: Discrete Distribution of Perturbations from the Bin-CVC Network Algorithm. A) $w = 5$ and $m = 2$, B) $w = 7$ and $m = 3$, C) $w = 8$ and $m = 5$ and D) $w = 12$ and $m = 6$.

After the network is set up with parameters w and m, the noise distribution D is fixed, and its mean μ and variances σ^2 are finite and known. Figure 5-2 showed this property. We intend to bound the noise e_q drawn from D in terms of μ and σ^2. We will continue to discuss how to estimate the mean μ and variances σ^2 in the next chapter.

5.3.3 Discussion

For Bin-CVC, there is a conflict between the two performance measures, CB and Z -score. That is, a high Column Balancing value, which indicates a good protection for the whole database with some specific w and m, could not guarantee good query answers (i.e., a high Z value).

We claim that *Interval disclosure* or *interval inference* occurs when the maximum of the error of the snooper's estimation about the true confidential value is less than the tolerance threshold predetermined by the DBA. *Exact inference* can be treated as a special case of interval inference and has an error value of 0.

Gopal et al. (2002) state that the CVC technique could completely eliminate exact disclosure and interval inference. However, Muralidhar et al. (2004) have shown empirically that CVC technique is sometimes vulnerable to interval inference. By utilizing a simple deterministic procedure, the snooper can sometimes compromise the database by shrinking the interval answers into a smaller range within the predetermined threshold. Suppose the i^{th} query is answered by $[l_i, u_i]$. In their example, they show how a snooper could compute the midpoint of the interval $m_i = (l_i + u_i)/2$, the half-width of the interval, $w_i = (u_i - l_i)/2$, and then use these to build a new interval as $m_i \pm (0.5 \times w_i)$ which still includes the true value, but is narrower than the original interval and, hence, less than the threshold. See Table 5-3 for this example.

Table 5-3: An Example of Interval Disclosure (Data source: Muralidhar et al. 2004)

Query	True Value	P^1	P^2	P^3	Original Interval			Intruder Interval		
					Lower Limit	Upper Limit	Width of Original Interval (%)	Lower Limit	Upper Limit	Width of Modified Interval (%)
1	276.3	275.2	302.8	263.5	263.5	302.8	14.2	273.3	293.0	7.1
2	35.4	36.2	32.7	36.3	32.7	36.2	10.2	33.6	35.4	5.1
3	37.4	37.4	41.1	35.5	35.5	41.1	14.9	36.9	39.7	7.5
…	…	…	…	…	…	…	…	…	…	…

In Gopal et al. (2002), the interval protection requires that the interval length is at least 10% of the original value. In Table 5-3, the intruder's intervals computed using the method provided by Muralidhar et al. (2004) are narrower than the threshold of 10%. Thus, the database is compromised in terms of the interval disclosure.

However, the test given by Muralidhar et al. (2004) only examined the CVC interval protection empirically. For networks with different w and m, this deterministic method may not apply.

5.4 A Bound for The Fixed-data Perturbation (Theoretical Basis)

Dinur and Nissim (2003) studied a theoretical tradeoff between privacy and usability of statistical databases (SDBs). They concluded that a minimum perturbation magnitude of $\Omega\left(\sqrt{n}\right)$ is required for each query q in order to maintain even weak privacy of the database. Otherwise, an adversary could reconstruct the statistical database using $l = n\left(\lg n\right)^2$ (base 2 logarithm) queries with high probability in polynomial time. As expected, the SDB can be protected from disclosure if the perturbation value is bounded by $e > o\left(\sqrt{n}\right)$, however, then the data utility may be too low to be useful. Since Dinur and Nissim make no assumptions beyond assuming the additive error is fixed, their results are valid both for data perturbation and output perturbation methods using fixed additive error. We review their results and methodology in the following sections.

Dinur and Nissim (2003) modeled the confidential field in the database as an n-bit binary string $(d_1,...,d_n) \in \{0,1\}^n$. The true answer for a SUM query q, $q \subseteq \{1,\cdots,n\}$, is computed as $\sum_{i \in q} d_i$. The perturbed answer for a query q is $A(q)$ obtained by adding a perturbation $\left| A(q) - \sum_{i \in q} d_i \right| \le e$, where $e = o(\sqrt{n})$ is the bound for the perturbation of each query.

The authors developed a Linear Programming (LP) algorithm to generate the candidate confidential vector which is the vector that an adversary would use to compromise the database. See Table 5-4 for details of the LP algorithm.

Table 5-4: LP Algorithm (Data source: Dinur and Nissim 2003).

[Query Phase]

Let $l = n(\lg n)^2$. For $1 \le j \le l$ choose uniformly at random $q_j \subseteq \{1,\cdots,n\}$, and set $\tilde{a}_{q_j} \leftarrow \mathrm{A}(q_j)$.

[Weeding Phase]

Using and linear objective, solve the following linear program with unknowns c_1, …, c_n:

$$\tilde{a}_{q_j} - e \le \sum_{i \in q_j} c_i \le \tilde{a}_{q_j} + e \qquad \text{for } 1 \le j \le l$$

$$0 \le c_i \le 1 \qquad \text{for } 1 \le i \le n$$

[Rounding Phase]

Let $c_i' = 1$ if $c_i > 1/2$ and $c_i' = 0$ otherwise.

Output c'.

Other vectors that are far away from the true confidential vector d are weeded out by the algorithm. The output of the LP algorithm is the candidate vector that best estimates the confidential vector.

The n-bit binary vector c' is obtained by rounding c, which is a vector of real numbers produced by the LP algorithm. Dinur and Nissim (2003) also introduced a vector $\overline{c}$ obtained by rounding c to the nearest integer multiple of $\frac{1}{k}$, where $k = n$ represents a precision parameter, and $K = \left\{0, \frac{1}{k}, \frac{2}{k}, ..., \frac{k-1}{k}, 1\right\}$. Hence $\overline{c} \in K^n$. They proved that $\left|\sum_{i \in q_j} \left(\overline{c}_i - d_i\right)\right| \leq 2e + 1$.

To prove that the candidate vector c' obtained from the algorithm is close to the true confidential field d, Dinur and Nissim (2003) introduced a Disqualifying Lemma, which proves that random queries $q_1, ..., q_l$ would weed out all vectors $x \in X$ where

$$X = \left\{ x \in K^n \mid \Pr_i \left[\left| |x_i - d_i| \geq \frac{1}{3} \right| \right] > \varepsilon n \right\} \qquad (1)$$

The term $\Pr_i \left[\left| |x_i - d_i| \geq \frac{1}{3} \right| \right]$ in Equation 1 represents the expected number of records that obey $|x_i - d_i| \geq \frac{1}{3}$, for $\varepsilon > 0$. Therefore, X denotes the set of all vectors which are far away from the true vector d.

The Disqualifying Lemma states that

$$\Pr_{q \subseteq_R [n]} \left[\left| \sum_{i \in q} (x_i - d_i) \right| \geq 2e + 1 \right] > \xi \qquad (2)$$

The lemma proves that there exists a probability $\xi > 0$ such that a query q disqualifies x if $\left|\sum_{i \in q}(x_i - d_i)\right| \geq 2e+1$. x will not be a valid LP solution if such a q exists. The lemma guarantees if x is far away from d, at least one of the l queries $q_{1,} \cdots, q_l$ would disqualify x with high probability.

One missing piece is the relationship between inequalities (1) and (2) that relates ε to ξ. The proof of the disqualifying lemma establishes this link and it is possible to think of ξ as a function of ε: $\xi(\varepsilon)$. We will discuss this further in Chapter 6.

If l queries $q_{1,} \cdots, q_l$ are chosen independently and randomly, then for each $x \in X$, the probability that all l queries do not disqualify x is $(1-\xi)^l$.

A conclusion derived from the Disqualifying Lemma is

$$\Pr_{q_i, \ldots q_l \subseteq_R [n]}\left[\forall x \in X \ \ \exists i, \ q_i \ \ disqualifies \ \ x\right] \geq 1-(n+1)^n (1-\xi)^l \geq 1-neg(n)$$

→

$$1-\Pr_{q_i, \ldots q_l \subseteq_R [n]}\left[\forall x \in X \ \ \exists i, \ q_i \ \ disqualifies \ \ x\right] \leq (n+1)^n (1-\xi)^l \leq neg(n)$$

Thus, the probability that none of the l queries can disqualify $x \in X$ is bounded by a very small number $neg(n) > 0$.

Therefore, the Disqualifying Lemma guarantees ruling out all disqualifying vectors $x \in X$ with high probability ($1-neg(n)$) and guarantees that the hamming distance between the final candidate vector c' and true vector d is small, that is, $dist(c', d) \leq \varepsilon n$.

The number of queries that are required to weed out disqualified vectors is computed from the Disqualifying Lemma. That is, $l = n(\lg n)^2$. See Figure 5-3 for an illustration of relationships of c, c', $\bar{c}$ and d.

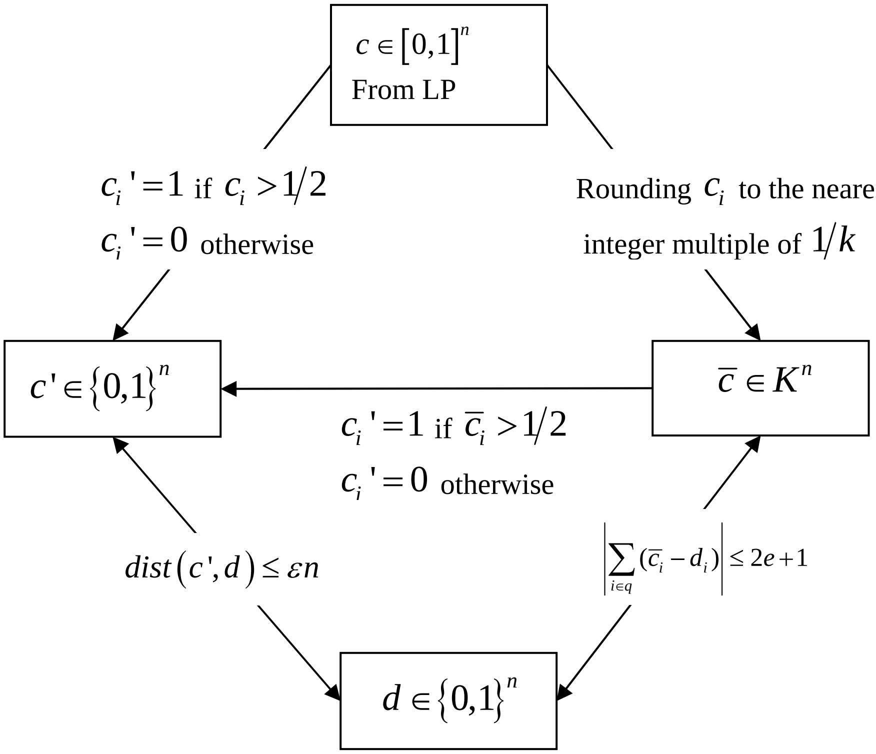

$c \in [0,1]^n$
From LP
$c_i' = 1$ if $c_i > 1/2$
$c_i' = 0$ otherwise
Rounding c_i to the nearest integer multiple of $1/k$
$c' \in \{0,1\}^n$
$\bar{c} \in K^n$
$c_i' = 1$ if $\bar{c}_i > 1/2$
$c_i' = 0$ otherwise
$dist(c', d) \le \varepsilon n$
$\left|\sum_{i \in q} (\bar{c}_i - d_i)\right| \le 2e + 1$
$d \in \{0,1\}^n$

Figure 5-3: Relationships of c, c', $\bar{c}$ and d.

5.5 Proposed Approach

Although SDC methods and machine learning have completely opposite research goals, similar methodologies are applied in both areas (Domingo-Ferrer and Torra 2003). The SDC methods attempt to modify the data intentionally before the public release. The data distortion should be sufficient enough to protect the privacy of the confidential data and small enough to minimize the information loss. ML seeks to learn from noisy examples and designs error-resilient algorithms to disclose true information (Angluin and Laird 1988, Goldman and Sloan 1995, Shackelford and Volper 1988, Sloan 1988, Valiant

1985). SDC methods protect the confidential data stored in a database with n records and m fields. ML learns the true function from l examples, each of them having m attributes. Therefore, a common structure is used to express the information between SDL methods and ML. Although the two areas have different research purpose and often use different terminologies, the underlying methodologies are often the same.

In our research, we approach the database privacy problem from a machine learning perspective by applying PAC learning theory. We consider a scenario when a snooper uses a learning algorithm to discover the true confidential data protected by a SDC method. For example, Figure 5-4 demonstrates the connection between the methodologies employed in PAC learning theory and in the database protection approach in Dinur and Nissim (2003).

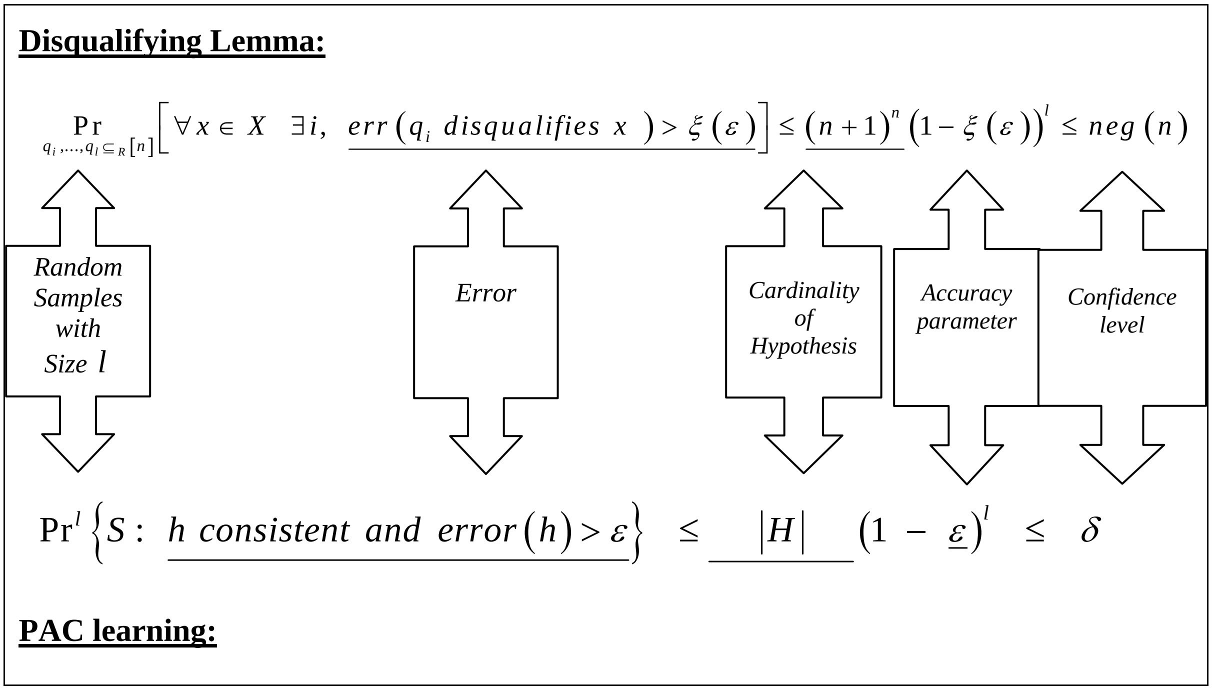

Disqualifying Lemma:
$\Pr_{q_i,\ldots,q_l \subseteq_R [n]}\left[\forall x \in X \ \exists i,\ \underline{err\left(q_i\ disqualifies\ x\right) > \xi(\varepsilon)}\right] \le \underline{(n+1)^n}\left(1-\xi(\varepsilon)\right)^l \le neg(n)$
Random Samples with Size l
Error
Cardinality of Hypothesis
Accuracy parameter
Confidence level
$\Pr^l\left\{S:\ \underline{h\ consistent\ and\ error(h) > \varepsilon}\right\} \le \underline{|H|}\ (1 - \underline{\varepsilon})^l \le \delta$
PAC learning:

Figure 5-4: Illustration of the Connection between the PAC Learning and Data Perturbation

Figure 5-4 indicates that both approaches determine a training sample size l, necessary to accomplish the desired goal. The probability that a query disqualifies the

$x \in X$ with probability greater than $\xi(\varepsilon)$ is bounded by the union bound of X, high probability $\xi(\varepsilon)$, and further bounded by a small probability $neg(n)$. Those three parameters correspond to the cardinality of the hypothesis space $|H|$, the accuracy parameter ε, and the confidence level δ in the PAC learning theory. They are shown in Figure 5-4 as matched terms even though different notation and terminologies are adopted. Therefore, we could conclude that both PAC learning theory and the Disqualifying Lemma address the problems by using the same methodology for different purposes. The same parameters are required to build up the models.

From the perspective of PAC learning theory, we regard the true confidential field as the target concept that an adversary seeks to discover within a limited number of queries in the presence of some noise, such as random data perturbation or variable-data perturbation. In Chapter 6, we raise our research questions and extend Dinur and Nissim (2003)'s work by using PAC learning theory. We set up a model to describe how much protection is necessary to guarantee that the adversary cannot discover the database with high probability. Put in PAC learning terms, we derive bounds on the amount of error an adversary makes, given a general perturbation scheme, the number of queries, and a confidence level.

Three types of data perturbation bounds are summarized as follows in terms of different error distributions.

(1) Perturbation with a General Bound Case: General PAC bound

The error is randomly generated identically and independently from an unknown distribution D. So it is also called Perturbation with a Distribution-free Bound case. A general PAC bound is derived as:

$$l \geq \frac{1}{\varepsilon}\left(\ln\left|H\right| + \ln\frac{1}{\delta} \right)$$

where l is the number of queries needed to discover the binary confidential data, ε is the amount of error that an adversary may make to compromise the database and δ is the confidence level. $\left|H\right| = 2^n$ is the number of candidate confidential vectors in the hypothesis space H. Without specific information about the distribution of noise, the derivation of l wholly depends on ε and δ, so this bound is relatively loose.

(2) Perturbation with a Fixed-data Bound Case: Fixed data perturbation

Dinur and Nissim (2003) derived a fixed-data bound $e = o\left(\sqrt{n}\right)$ for the perturbation added to query responses. A bound for the number of queries is also developed, denoted as:

$$l = n\left(\lg n\right)^2$$

which is sufficient to discover the true confidential vector in the database with a high probability at a small error.

(3) Perturbation with a Random Variable Bound Case: Variable data perturbation (Proposed research)

We assume that random perturbations which are added to the query responses have an unknown discrete distribution. The moments of the distribution, such as the mean and standard deviation, can be estimated. Variable-data perturbation belongs to this case. In the next chapter, we derive an error bound for this case by applying the PAC learning theory. This bound provides the minimum number of queries needed to discover the protected column with specified error and accuracy.

CHAPTER 6
DISCLOSURE CONTROL BY APPLYING LEARNING THEORY

In Chapter 2 and 3 we reviewed PAC learning theory and database security methods. In this chapter, we approach the database privacy problem using ideas from Probably Approximate Correct learning theory. Our research will delve into the additive noise perturbation masking method which is classified into three categories: random data perturbation, fixed data perturbation (reviewed in Chapter 5) and variable-data perturbation. Based on the work of Garfinkel et al. (2002) and Dinur and Nissim (2003), we raise our research questions and construct a theoretical model from the perspective of PAC learning theory. We attempt to derive an error bound for perturbations with a distribution specified by its first two moments and also develop a heuristic method to estimate the mean and standard deviation for the variable-data perturbation method. Dinur and Nissim (2003) studied the case of data perturbation bounded by a fixed number and provide a theoretical foundation for our research.

6.1 Research Problems

Our research focuses on the category of variable-data perturbation. Firstly, we intend to derive a bound on the level of error that an adversary may make, given the variable-data perturbation method. We extend the bound on the fixed-data perturbation proposed by Dinur and Nissim (2003) with an attempt to bound the perturbation of each query with a random variable e_q which has a discrete distribution with known parameters, such as the finite mean and variance. We need to develop a new

Disqualifying Lemma, analogous to Dinur and Nissim's (2003), for the variable-data perturbation by deploying PAC learning theory. Like the Disqualifying Lemma in Dinur and Nissim (2003), our result bounds the probability that a query does not eliminate hypotheses that are far away from the true confidential answer. Using this, we develop an error bound on the number of queries within which the database could be compromised with high probability.

6.2 The PAC Model For the Fixed-data Perturbation

We start our model by interpreting the results of Dinur and Nissim (2003) within the methodology of PAC learning theory.

Suppose an adversary attempts to compromise the SDB by applying PAC learning theory. We define a *Non-Private Database* as follows: a database is non-private if a computationally-bound adversary can expose $1-\varepsilon$ fraction of the confidential data for $\varepsilon > 0$ with probability $1-\delta$, where $\delta > 0$. We call $1-\delta$ the confidence level.

Consider a statistical database with n records. Its confidential field is a binary string denoted as $(d_1,...,d_n)' \in \{0,1\}^n$. See Table 5-1 for an example database. In this table, "HIV" status is the column we represent. An hypothesis space H_0 contains n-bit binary vectors, each of which is an hypothesis $h \in H_0 = \{0,1\}^n$ and denotes a candidate vector for the confidential field of the database. The cardinality of the hypothesis space, or the number of hypothesis is $|H_0| = 2^n$. The true confidential field is regarded as the target concept $d \in H_0$. The online database receives a SUM (or COUNT) query $q \subseteq \{1,...,n\}$ sent by the user and responds with a perturbed answer $A(q)$ of the true answer $a_q = \sum_{i \in q} d_i$. A perturbation is added to each query answer instead of every record and bounded by a fixed number $e \geq \left| a_q - A(q) \right|$.

PAC Learning starts by random sampling. We take l samples consisting of queries and their perturbed responses,

$$S = \left(\left(q_1, A(q_1) \right), \cdots, \left(q_l, A(q_l) \right) \right).$$

Since $A(q)$ is a perturbed answer, we will consider this learning from noisy data.

Our learning algorithm is a linear program. As such, answers can be continuous and will be rounded. Thus it is useful to define another hypothesis space $H_2 = [0,1]^n$. For analysis, a grid will prove useful. Let the hypothesis space $H_1 = K^n$, where $K = \left\{ 0, \frac{1}{n}, \frac{2}{n}, ..., \frac{n-1}{n}, 1 \right\}$. Note that $H_0 \subseteq H_1 \subseteq H_2$ where all containments are strict when $n > 1$.

Let $h_1 : H_2 \to H_1$ by rounding each component in H_2 to the nearest integer multiple of $1/n$ (midpoints rounded down). Further, let $h_0 : H_i \to H_0 \quad (i = 1,\ 2)$ by rounding each component in H_i to the nearest of 0 and 1 (0.5 rounds down). Note that $h_1(c) = c + f$, where $|f_i| < \frac{1}{n} \qquad i = 1, \cdots, n$.

Given a sample S and a fixed perturbation e, Dinur and Nissim (2003) gave a polynomial algorithm γ that finds $c \in H_2$, from which one can output $h_0(c)$. We represent this algorithm by $c \leftarrow \gamma(S)$. As already discussed, the specific algorithm is a linear program (see Table 5-4).

See Figure 6-1 for an illustration of the relationships of H_0, H_1, H_2, h_0, h_1 and d.

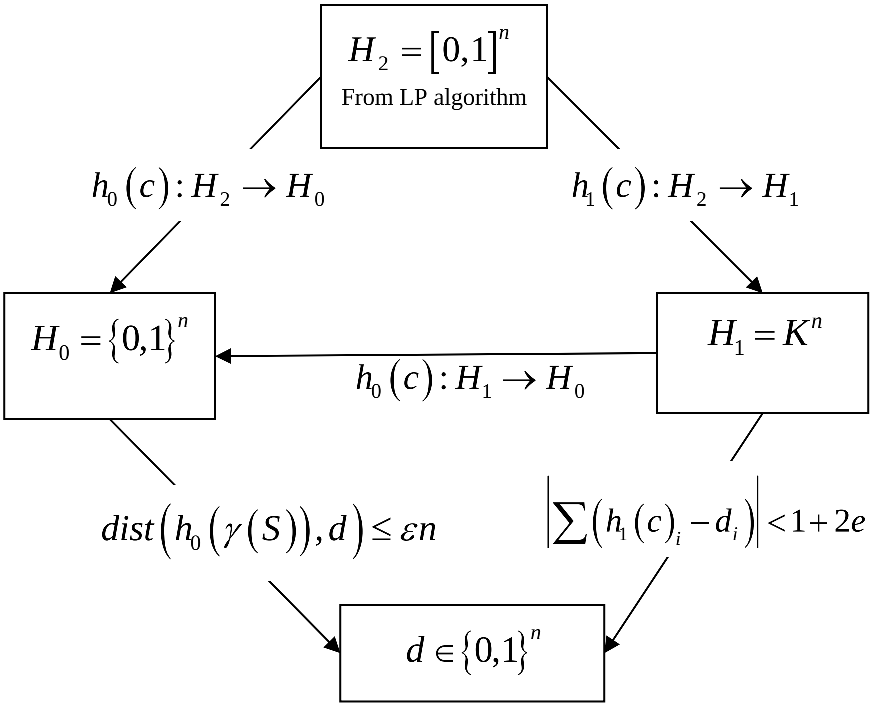


Figure 6-1: Relationships H_0, H_1, H_2, h_0, h_1 and d in the Fixed-Data Perturbation.

Let $c \in H_0$, then the hamming distance between c and d is

$$dist(c,d) = \left|\{i : c_i \neq d_i\}\right| = \sum_{i=1}^{n} |c_i - d_i|.$$

Let $x \in H_2$. $\Pr_i\left[|x_i - d_i| \geq \frac{1}{3}\right] > \varepsilon$ means the probability of choosing $i \in \{1, \cdots, n\}$ randomly such that $|x_i - d_i| \geq \frac{1}{3}$. That is, for this x there are εn expected records where $|x_i - d_i| \geq \frac{1}{3}$. Denote this by $E_i\left[|x_i - d_i| \geq \frac{1}{3}\right] > \varepsilon n$ where $\varepsilon > 0$ arbitrarily. Ultimately, we wish to show how to choose a sample size l so that $dist\left(h_0(\gamma(S)), d\right) \leq \varepsilon n$.

Lemma 1:

If $x \in K^n$ and $E_i\left[|x_i - d_i| \geq \frac{1}{3}\right] \leq \varepsilon n$, then $dist\left(h_0(x), d\right) \leq \varepsilon n$

Proof:

First note that if $\left|x_i - d_i\right| \le \frac{1}{3}$ then $\left|h_0\left(x\right)_i - d_i\right| \le \frac{1}{2} - \frac{1}{3} < \frac{1}{3}$. Thus since no more than εn i's, on average, have $\left|x_i - d_i\right| \ge \frac{1}{3}$, then no more than εn records, on average, of $h_0\left(x\right)$ can have $\left|h_0\left(x\right)_i - d_i\right| \ge \frac{1}{3}$. The number $\frac{1}{3}$ in $\left|x_i - d_i\right| \le \frac{1}{3}$ guarantees that x_i round to the same number as d_i.

End of Proof

Let $T = \left\{ x \in K^n : \ E_i\left[\left|x_i - d_i\right| \ge \frac{1}{3}\right] > \varepsilon n \right\}$. From the point of view of the intruder, we want our sample to disqualify all points of T with high probability $\left(1-\delta\right)$ where $\delta \in \left(0,1\right)$ and is usually chosen so that $\left(1-\delta\right)$ is large. For a sample of size l, generated independently and identically according to an unknown but fixed distribution D, the probability that an hypothesis c is far away from the true target d is measured by the risk functional

$$\begin{aligned}\underset{D}{err}\left(c\right) &= 1 - D\left(q \subseteq \left\{1,\cdots,n\right\} : \left|\sum_{i\in q}\left(c_i - d_i\right)\right| \le 1 + 2e \right)\\ &= D\left(q \subseteq \left\{1,\cdots,n\right\} : \left|\sum_{i\in q}\left(c_i - d_i\right)\right| > 1 + 2e \right)\end{aligned}$$

where $c \in H_1$.

As we stated before (see Figure 6-1), the solution c from the LP can be rounded either to a binary vector $h_0\left(c\right)$ or a vector $h_1\left(c\right) \in K^n$. The probability that the distance between the true vector d and the rounded vector $h_1(c)$ is greater than $\frac{1}{3}$ is bounded by ε. Based on this condition, for any random query, the difference between the answers from these two vectors is bounded by a function of the perturbation, $2e+1$. So, we can see that e and ε are related and they describe the error from different perspectives. Then we use a probability ξ which is a function of ε, denoted as $\xi(\varepsilon)$, to bound the risk functional as

$$\underset{D}{err}(c) > \xi(\varepsilon)$$

We intend to bound

$$D^l\left(S : \underset{D}{err}\left(h_1(\gamma(S))\right) > \xi(\varepsilon)\right)$$

by $\delta > 0$.

Provided $e = o\left(\sqrt{n}\right)$, the Disqualifying Lemma of Dinur and Nissim (2003) proved $\xi(\varepsilon) > 0$. Then, for $\kappa(\varepsilon) = 1 - \xi(\varepsilon)$

$$D^l\left(S : \underset{D}{err}\left(h_1(\gamma(S))\right) > \xi(\varepsilon)\right) \leq (n+1)^n \left(1 - \xi(\varepsilon)\right)^l = (n+1)^n \kappa^l(\varepsilon) \qquad (6.1)$$

where $(n+1)^n = |K| \geq |T|$ is the union bound over T, and therefore the worst-case scenario is bounded.

The proof of the Disqualifying Lemma in Dinur and Nissim (2003) shows

$$\kappa(\varepsilon) \leq \min\left(\underbrace{1 - \left(1 - 2e^{-T^2/8}\right)}_{(1)},\ \underbrace{1 - \frac{\alpha}{3\beta}}_{(2)}\right)$$

with $T \geq \sqrt{\frac{\varepsilon}{500}}$.

Recall that the Disqualify Lemma (Dinur and Nissim 2003) proves

$$\Pr_{q\subseteq_R[n]}\left[\left|\sum_{i\in q}(x_i-d_i)\right|\geq 2e+1\right]>\xi$$

In the proof, $\varpi_1,\varpi_2,\cdots,\varpi_n$ are defined as independent random variables such that $\varpi_i = x_i - d_i$ and $\varpi_i = 0$ both with probability $\frac{1}{2}$. Let $\varpi=\sum_{i=1}^{n}\varpi_i$. The authors approached the proof by dividing it into two cases based on the size of the expected value of ϖ, denoted as $E(\varpi)$. Let $T\geq\sqrt{\frac{\varepsilon}{500}}$ be a constant to be specified later in the proof.

In the case of $E(\varpi)\geq T\sqrt{n}$, the probability satisfies

$$\Pr_{q\subseteq_R[n]}\left[\left|\sum_{i\in q}(x_i-d_i)\right|\geq 2e+1\right]\geq 1-2e^{-T^2/8}$$

In the second case of $E(\varpi)<T\sqrt{n}$, the probability satisfies

$$\Pr_{q\subseteq_R[n]}\left[\left|\sum_{i\in q}(x_i-d_i)\right|\geq 2e+1\right]\geq\frac{\alpha}{3\beta}$$

The role of β is discussed below. (For the proof details, see the Appendix A of Dinur and Nissim 2003).

From the result of Disqualifying Lemma, we choose $\kappa(\varepsilon)$ to be the minimum of the probabilities from these two cases. So, in term (1), $1-2e^{-T^2/8}=1-2e^{-\varepsilon/4000}<0$, so $1-\left(1-2e^{-T^2/8}\right)>1$. In term (2), we know $\alpha=\frac{\varepsilon}{36}$, so $\frac{\alpha}{3\beta}=\frac{\varepsilon}{108\beta}\geq 0$ and $1-\frac{\alpha}{3\beta}<1$.

Hence

$$\kappa(\varepsilon) \le \min\left(\underbrace{1-\left(1-2e^{-T^2/8}\right)}_{(1)},\ \underbrace{1-\frac{\alpha}{3\beta}}_{(2)} \right) = 1-\frac{\alpha}{3\beta}$$

Thus,

$$\xi(\varepsilon) \ge \frac{\alpha}{3\beta}$$

where we choose $\xi(\varepsilon) = \frac{\alpha}{3\beta}$ for the worst case. Dinur and Nissim choose β large enough so that

$$\alpha > 3\sum_{k=1}^{\infty} \beta(k+1)e^{-k\beta/2}$$

(note the right side is decreasing in β). Simple manipulations show that

$$\sum_{k=1}^{\infty} e^{-k\beta/2} = \frac{e^{-\beta/2}}{1-e^{-\beta/2}}$$

After taking the partial derivative with respects to β for the above formula we obtain

$$-2\frac{\partial}{\partial\beta}\sum_{k=1}^{\infty} e^{-k\beta/2} = \sum_{k=1}^{\infty} ke^{-k\beta/2} = \frac{e^{-\beta/2}}{\left(1-e^{-\beta/2}\right)^2}$$

Thus

$$\sum_{k=1}^{\infty}(k+1)e^{-k\beta/2} = \frac{e^{-\beta/2}}{\left(1-e^{-\beta/2}\right)^2} + \frac{e^{-\beta/2}}{1-e^{-\beta/2}} = e^{-\beta/2}\frac{2-e^{-\beta/2}}{\left(1-e^{-\beta/2}\right)^2}$$

Thus we need

$$\alpha > 3\sum_{k=1}^{\infty} \beta(k+1)e^{-k\beta/2} = 3\beta e^{-\beta/2}\frac{2-e^{-\beta/2}}{\left(1-e^{-\beta/2}\right)^2}$$

Since $\alpha = \frac{\varepsilon}{36}$, we get

$$\frac{\varepsilon}{36} > 3\beta e^{-\beta/2} \frac{2-e^{-\beta/2}}{\left(1-e^{-\beta/2}\right)^2}$$

$$\varepsilon > 108\beta e^{-\beta/2} \frac{2-e^{-\beta/2}}{\left(1-e^{-\beta/2}\right)^2}$$

Let $x = e^{-\beta/2}$. Then

$$\varepsilon > 108\beta x \frac{2-x}{(1-x)^2}$$

β is decided by ε (ε is a pre-defined parameter). For $0 < \varepsilon < 1$, numerical calculations show we need $\beta > 17$ thus giving $x < 0.0002$. Since

$$\kappa(\varepsilon) \le 1 - \frac{\alpha}{3\beta} = 1 - \frac{\varepsilon}{108\beta} \ ,$$

if we plug

$$\varepsilon > 108\beta x \frac{2-x}{(1-x)^2}$$

into

$$\kappa(\varepsilon) \le 1 - \frac{\alpha}{3\beta} = 1 - \frac{\varepsilon}{108\beta},$$

we get

$$\kappa(\varepsilon) \le 1 - x\frac{2-x}{(1-x)^2}$$

where $\xi(\varepsilon) = \dfrac{\alpha}{3\beta} = x\dfrac{2-x}{(1-x)^2}$.

Now back to the inequality (6.1),

$$D^l\left(S : \underset{D}{err}\left(h_1\left(\gamma\left(S\right)\right)\right) > \xi\left(\varepsilon\right)\right) \le \left(n+1\right)^n \kappa^l\left(\varepsilon\right)$$

$$=\left(n+1\right)^{n}\left(1-\xi\left(\varepsilon\right)\right)^{l}.$$

If we bound the probability with the parameter $\delta > 0$, we get

$$D^{l}\left(S:\underset{D}{err}\left(h_{1}\left(\gamma\left(S\right)\right)\right)>\xi\left(\varepsilon\right)\right)\leq\left(n+1\right)^{n}\left(1-\xi\left(\varepsilon\right)\right)^{l}\leq\delta$$

where $\delta > 0$ is the confidence parameter.

Then take the base 2 logarithm (denoted as $\lg$ in all the following formula) on both sides of the last two terms

$$\left(n+1\right)^{n}\left(1-\xi\left(\varepsilon\right)\right)^{l}\leq\delta$$

to get

$$\lg\left[\left(n+1\right)^{n}\left(1-\xi\left(\varepsilon\right)\right)^{l}\right]\leq\lg\delta$$

Given a pre-defined parameter ε, the minimum sample size is computed as

$$l\geq\frac{\lg\left(\delta\right)-n\lg\left(n+1\right)}{\lg\left(1-\xi\left(\varepsilon\right)\right)}\qquad(6.2)$$

where $\xi\left(\varepsilon\right)=x\dfrac{2-x}{\left(1-x\right)^{2}}$, and $x=e^{-\beta/2}$ with β chosen large enough. l is bounded by three parameters $\delta,\ \varepsilon$ and n. Since $\xi\left(\varepsilon\right)$ is a very small number, if we apply it directly into formula (6.2), the resulting bound for the sample size l is quite large, much more than $l=n\left(\log n\right)^{2}$ from Dinur and Nissim (2003), even for a small n. See Table 6-1 for examples of two bounds on the sample size with different values of n when $\delta=0.05$.

Table 6-1 shows that by interpreting Dinur and Nissim (2003)'s Disqualifying Lemma, we get a PAC bound which is looser than the one derived in Dinur and Nissim

(2003), no matter what n is. However this PAC bound is still much less than the total number of queries in a database, 2^n, except the n is very small, such as $n=10$.

Table 6-1: Bounds on the Sample Size with Different Values of n.

n	$l = n(\log n)^2$	$l \geq \frac{\lg(\delta) - n\lg(n+1)}{\lg(1-\xi(\varepsilon))}$	2^n
10	111	373643	1024
50	1,593	2,274,447	1.1259E+15
100	4,415	5,191,750	1.2677E+30
500	40,193	34,338,167	3.2734E+150
1000	99,317	76,188,677	1.0715E+301
5000	754,940	469,076,527	---

In section 6.4, we will show how to replace $\xi(\varepsilon) = x\frac{2-x}{(1-x)^2}$ with a more practical number by using the bound in Dinur and Nissim (2003), therefore deriving a tighter bound for the variable-data perturbation case.

6.3 The PAC Model For the Variable-data Perturbation

In this section, we move to the case that an adversary tries to compromise a database in which the confidential data is modified by adding variable-data perturbation. In this method, each query q is added with a perturbation created from a database protection algorithm. The perturbed response is $A(q)$ while the true query answer is $a_q = \sum_{i\in q} d_i$.

6.3.1 PAC Model Setup

In the fixed-data perturbation case, a fixed number bounds the perturbation: $|a_q - A(q)| \leq e$. In the variable-data perturbation case, $|a_q - A(q)| = e_q$ and we assume that the perturbation e_q is a random variable with an unknown discrete distribution with known finite mean μ and variance σ^2. Based on the knowledge of these parameters, we attempt to develop a bound on the error that an adversary makes. The bound will be expressed in terms of these parameters. A threshold on the number of queries, within which the database is compromised, can be derived from this error bound.

Given S and $\tilde{e}_q$ for each $q \in S$, we develop a polynomial algorithm γ_2 that obtains an hypothesis $c \in H_2$ from which we can output $h_0(c)$. The algorithm, $c \leftarrow \gamma_2(S)$, is a linear program:

$$\underset{c_i \in [0,1]}{Min} \sum_{i=1}^{n} c_i$$

$$\text{s.t.} \qquad \left| \sum_{i \in q_j} c_i - A(q_j) \right| \le \tilde{e}_{q_j} \qquad i = 0, \cdots, n \qquad j = 1, \cdots, l$$

where $\tilde{e}_{q_j}$ is the realization of the random variable e_q in the LP algorithm and is sampled from the perturbation distribution. Then the distance between $h_1(c)$ and the true vector d is bounded by

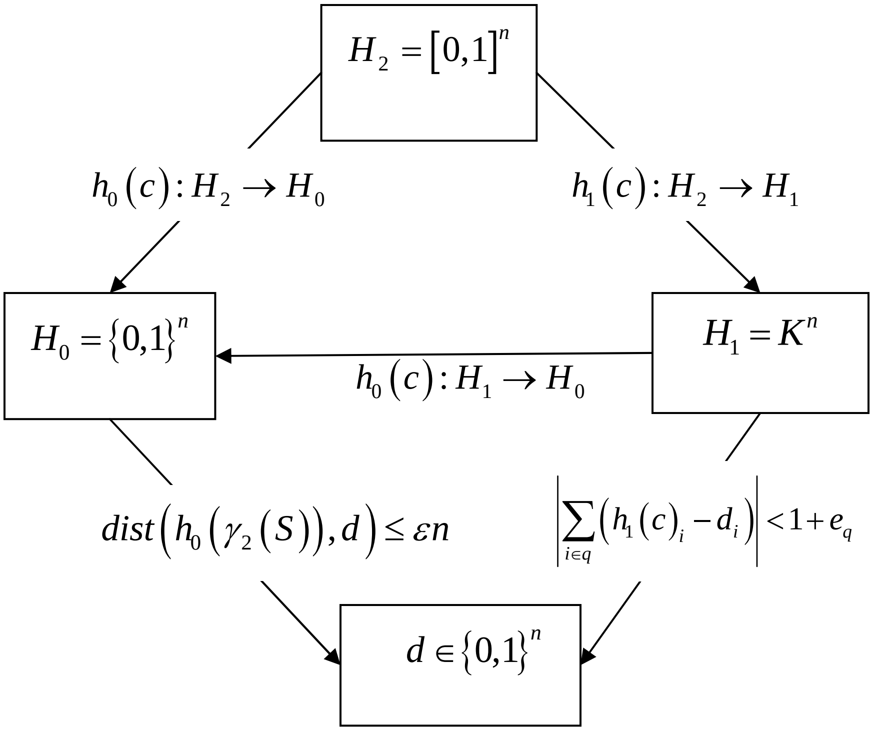
$$\begin{aligned} \left| \sum_{i \in q} \left(h_1(c)_i - d_i \right) \right| &\le \left| \sum_{i \in q} \left(h_1(c)_i - c_i \right) \right| + \left| \sum_{i \in q} (c_i - d_i) \right| \\ &\le \frac{|q|}{n} + e_q \\ &\le 1 + e_q \end{aligned}$$

where $h_1(c)_i = c_i + f_i$ and $|f|_i < \dfrac{1}{n}$. Recall that $|q|$ denotes the cardinality of the query q.

In the variable-data perturbation case, we need to develop a new Disqualifying Lemma which would disqualify all $h_1(c)$ which are far away from the true vector d.

That is, for any $x \in H_2$, query q disqualifies x, if $\left|\sum_{i\in q}\left(h_1(x)_i - d_i\right)\right| > 1 + e_q$.

See Figure 6-2 for an illustration of the relationships of H_0, H_1, H_2, h_0, h_1 and d.



$H_2 = [0,1]^n$
$h_0(c): H_2 \to H_0$
$h_1(c): H_2 \to H_1$
$H_0 = \{0,1\}^n$
$H_1 = K^n$
$h_0(c): H_1 \to H_0$
$dist\left(h_0\left(\gamma_2(S)\right), d\right) \le \varepsilon n$
$\left|\sum_{i\in q}\left(h_1(c)_i - d_i\right)\right| < 1 + e_q$
$d \in \{0,1\}^n$

Figure 6-2: Relationships of H_0, H_1, H_2, h_0, h_1 and d in the Variable-Data Perturbation

6.3.2 Disqualifying Lemma 2

For a sample of size l which is generated i.i.d according to an unknown but fixed discrete distribution D, the probability that an hypothesis $h_1(c)$ is far away from the true target d is measured by the risk functional

$$\underset{D}{err}\left(h_1(c)\right) = 1 - D\left(q \subseteq \{1, \cdots, n\} : \left|\sum_{i\in q}\left(h_1(c)_i - d_i\right)\right| \le 1 + e_q\right)$$

$$= D\left(q \subseteq \{1,\cdots,n\} : \left| \sum_{i \in q} \left(h_1(c)_i - d_i \right) \right| > 1 + e_q \right)$$

$$> \eta(\varepsilon)$$

We intend to bound this error rate. As in section 6.2, we want

$$D^l \left(S : \underset{D}{err} \left(h_1 \left(\gamma_2 (S) \right) \right) > \eta(\varepsilon) \right) \leq \delta \tag{6.3}$$

where $\delta \in (0,1)$.

We now develop our Lemma 2, a disqualifying lemma, analogous to Dinur and Nissim’s Disqualifying Lemma. Lemma 2 assumes that the mean and standard deviation of the distribution of e_q satisfies $\mu \geq \sigma$, $\sigma + \mu \leq 2\sqrt{n}$ and $\mu > \sqrt{n}$. Practical reasons motivate these respective cases as we now discuss.

(1) if $\mu \leq \sigma$:

Since the standard deviation measures how spread out the perturbations (e_q values) can be, if $\mu \leq \sigma$, many perturbations will be widely dispersed, meaning that the corresponding intervals offer little information. This can take many forms. For example (see Figure 6-3), with a bimodal distribution some intervals will be tight and others very disperse. The tight ones might provide an attacker the ability to easily disclose parts of the confidential information. The wide intervals may provide too little usable information to be meaningful for the user.

(2) if $\sigma + \mu \geq 2\sqrt{n}$, there are four possible cases:

a. $\mu \geq \sqrt{n} \geq \sigma$

In this case, most perturbations are clustered around a large mean. Although a large perturbation provides better protection of the database, it reduces the usability of the query answers. The user gets very little information. For a demonstration of this case, see the following Figure 6-4. Consequently, a database security method is meaningless if it produces perturbations with a large mean and relatively small standard deviation.

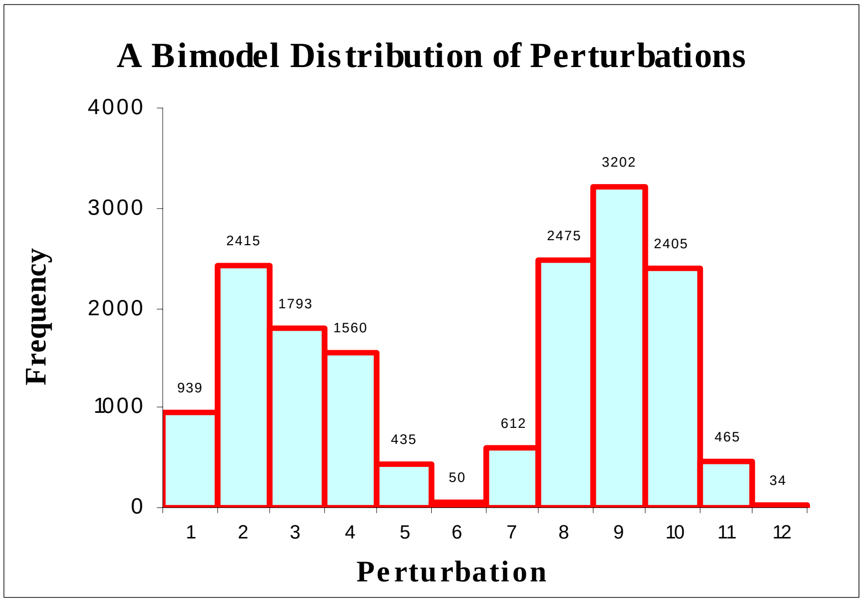


Figure 6-3: A Bimodal Distribution of Perturbations in the CVC Network while $\mu \le \sigma$.

b. $\mu \ge \sigma \ge \sqrt{n}$

Very high mean and standard deviation imply two situations: (1) all query responses are perturbed with big noises which are widely spread out in the high mean area. In this case, the user can not get any useful data from these query answers; and (2) many query answers have large perturbations while others provide users with very tight answers which can reveal the confidential data easily. Neither of above distributions is meaningful for our research.

c. $\sigma \ge \sqrt{n} \ge \mu$

The same reason described in (1) is used here also.

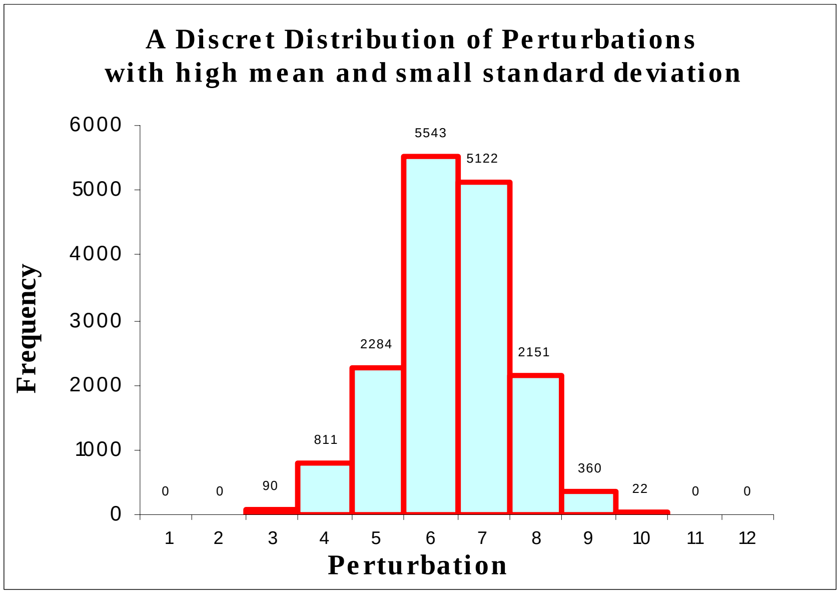


Figure 6-4: A Distribution of Perturbations in the CVC Network with $\mu \geq \sqrt{n} \geq \sigma$.

(3) if $\mu > \sqrt{n}$ holds:

A database usually includes a large number of records. Therefore, the mean of the perturbations is likely less than $\sqrt{n}$ in most cases. If the mean $\mu > \sqrt{n}$ is true, then the security method likely offers little information to the users, no matter what the standard deviation is. See the discussion in (2) a, b and c for similar explanations.

Lemma 2:

Let $x \in [0,1]^n, d \in \{0,1\}^n$ and e_q be a random variable generated from a distribution with mean $\mu = E(e_q) < \infty$ and variance $\sigma^2 < \infty$ where $\mu \geq \sigma$, $\sigma + \mu \leq 2\sqrt{n}$ and $\mu < \sqrt{n}$. If $\Pr_i\left[\left|h_1(x)_i - d_i\right| \geq \frac{1}{3}\right] > \varepsilon$, then there exists a constant $\eta(\varepsilon) > 0$, such that

$$\Pr_{q \subseteq R[n]}\left[\left|\sum_{i \in q}\left(h_1(x)_i - d_i\right)\right| > 1 + e_q\right] > \eta(\varepsilon)$$

where η is a function of ε.

Disqualifying Lemma 2 Proof:

Let $Y_i = h_1(x)_i - d_i$ be i.i.d. random variables. For any fixed $q \in [n]$, let $m = |q|$, the cardinality of q. Without loss of generality, assume $q = \{1,\cdots,m\}$. Given a random variable e_q, and constant $a \in \left[0,\sqrt{n}\right]$, we have

$$P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q\right) = P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q, e_q \le 2a\right) + P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q, e_q > 2a\right)$$

$$\ge P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q, e_q \le 2a\right)$$

$$= P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q \mid e_q \le 2a\right) P\left(e_q \le 2a\right)$$

$$= P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q \mid e_q \le 2a\right)\left(1 - P\left(e_q > 2a\right)\right)$$

According to Chebyshev's Inequality, since e_q is a random variable with $\mu = E\left(e_q\right) < \infty$, and $\sigma^2 < \infty$, then

$$P\left(e_q > 2a\right) = P\left(e_q - \mu > 2a - \mu\right) \le \frac{\sigma^2}{\left(2a - \mu\right)^2}$$

Then, we obtain

$$P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q\right) \ge \underbrace{P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q \mid e_q \le 2a\right)}_{(1)}\underbrace{\left(1 - \frac{\sigma^2}{\left(2a - \mu\right)^2}\right)}_{(2)} \qquad (6.4)$$

Let the probability $\eta(\varepsilon)$ be equal to the product of term (1) and term (2) in formula (6.4).

Next, we continue our proof by solving two problems, respectively.

(1) Prove $\eta(\varepsilon)$ is a positive number:

In all steps of Dinur and Nissim (2003)' proof for their Disqualifying Lemma, term (1) ,

$$P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q \mid e_q \leq 2a\right)$$

can be substituted for

$$P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + 2e\right)$$

provided $a \in \left[0, \sqrt{n}\right]$. To see this we have the following:

$$P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q \mid e_q \leq 2a\right) = \sum_{j=0}^{2a} P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + j\right) P\left(e_q = j\right)$$

$$\geq P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + 2e\right) \sum_{j=1}^{2a} P\left(e_q = j\right).$$

Since $E\left[e_q\right] \leq \sqrt{n}$, for any $a \geq \sqrt{n}$, $\sum_{j=0}^{2a} P\left(e_q = j\right) > 0$. Now, Dinur and Nissim (2003) proved $P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + 2e\right) \geq 1 - 2e^{-T^2/8}$ for the appropriate choice of T . Rescaling T in proportion to $\sum_{j=0}^{2a} P\left(e_q = j\right)$ proves our point. Similarly for the second part of his proof the parameters α and β can be rescaled in proportion to $\sum_{j=0}^{2a} P\left(e_q = j\right)$. This gives then that

$$P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q \mid e_q \leq 2a\right) \geq \max\left(1 - 2e^{-T^2/8}, \frac{\alpha}{3\beta}\right) = \frac{\alpha}{3\beta} = x\frac{2 - x}{\left(1 - x\right)^2}$$

where $x = e^{-\beta/2}$ with β chosen large enough as seen in Section 6.2. Thus

$$P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q\right) \geq x \frac{2-x}{(1-x)^2}\left(1 - \frac{\sigma^2}{(2a-\mu)^2}\right).$$

So the probability $\eta(\varepsilon)$ will be a positive number as long as term (2) is greater than 0. Thus we need to have

$$1 - \frac{\sigma^2}{(2a-\mu)^2} > 0$$

which is true when $0 \leq a \leq \frac{\mu-\sigma}{2}$ and $\frac{\sigma+\mu}{2} \leq a \leq \sqrt{n}$ provided $\mu \geq \sigma$ and $\sigma + \mu \leq 2\sqrt{n}$, respectively. These latter two conditions are assumed in the Lemma 2. Thus,

$$\Pr_{q \subseteq R[n]}\left[\left|\sum_{i \in q}\left(h_1(x)_i - d_i\right)\right| > 1 + e_q\right] > P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q \mid e_q < 2a\right)\left(1 - \frac{\sigma^2}{(2a-\mu)^2}\right) \quad (6.5)$$

where parameter $a \in \left[0,\ \frac{\mu-\sigma}{2}\right] \cup \left[\frac{\sigma+\mu}{2},\ \sqrt{n}\right]$.

(2) We now maximize the lower bound over a.

In order to derive a tight bound, we seek to find the maximum value of (6.5) subject to $a \in \left[0,\ \frac{\mu-\sigma}{2}\right] \cup \left[\frac{\sigma+\mu}{2},\ \sqrt{n}\right]$. So

$$P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q\right) \geq \max_a P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q \mid e_q \leq 2a\right)\left(1 - \frac{\sigma^2}{(2a-\mu)^2}\right)$$

$$\geq P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1 + e_q \mid e_q \leq 2a\right)\max_a\left(1 - \frac{\sigma^2}{(2a-\mu)^2}\right)$$

where the a in the first term is any $a \in \left[0,\ \frac{\mu-\sigma}{2}\right] \cup \left[\frac{\sigma+\mu}{2},\ \sqrt{n}\right]$. Using (for this term) a=o$\left(\sqrt{\text{n}}\right)$ gives us

$$P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1+e_q\right) \geq x\frac{2-x}{(1-x)^2}\max_a\left(1-\frac{\sigma^2}{(2a-\mu)^2}\right).$$

Note that

$$\left(1-\frac{\sigma^2}{(2a-\mu)^2}\right)$$

is decreasing over $\left[0,\ \frac{\mu-\sigma}{2}\right]$ and increasing over $\left[\frac{\sigma+\mu}{2},\ \sqrt{n}\right]$ so we merely need to compare

$$\left(1-\frac{\sigma^2}{\mu^2}\right)$$

to

$$\left(1-\frac{\sigma^2}{\left(2\sqrt{n}-\mu\right)^2}\right).$$

By assumption $\mu > \sqrt{n}$ so the latter is maximal. Thus

$$P\left(\left|\sum_{i=1}^{m} Y_i\right| > 1+e_q\right) \geq \eta(\varepsilon) \equiv x\frac{2-x}{(1-x)^2}\left(1-\frac{\sigma^2}{\left(2\sqrt{n}-\mu\right)^2}\right) > 0\,.$$

End of proof.

Lemma 2 is a crucial step for our model. The successful proof provides a bound on the error ε in terms of the mean and variances of e_q. In the next section, we will continue discussing these two parameters. Based on the results of Lemma 2, we are able to derive a bound for the number of queries, within which the adversary would be able to compromise the database protected by using the variable-data perturbation method with a high probability ($1-\delta$).

6.4 The Bound of the Sample Size for the Variable-data Perturbation Case

In this section, based on the proof of Lemma 2, we develop the sampling bound for the variable-data perturbation case from two approaches. In the first approach, we use Dinur and Nissim (2003)'s result directly from their Disqualifying Lemma proof in our bound; the second approach applies instead their sample bound to obtain a tighter bound.

6.4.1 The Bound Based On the Disqualifying Lemma Proof

Recall that $\underset{D}{err}\left(h_1\left(c\right)\right) > \eta\left(\varepsilon\right)$ (see section 6.3), and we intend to bound

$$D^l\left(S : \underset{D}{err}\left(h_1\left(\gamma\left(S\right)\right)\right) > \eta\left(\varepsilon\right)\right)$$

by the confidence parameter $\delta > 0$.

We use a probability $\chi\left(\varepsilon\right)$ to bound

$$\underset{D}{err}\left(h_1\left(c\right)\right) > \eta\left(\varepsilon\right).$$

Then,

$$D^l\left(S : \underset{D}{err}\left(h_1\left(\gamma_2\left(S\right)\right)\right) > \eta\left(\varepsilon\right)\right) \le \left(n+1\right)^n \chi^l\left(\varepsilon\right)$$

where $\chi\left(\varepsilon\right) = 1-\eta\left(\varepsilon\right)$. Thus we get

$$D^l\left(S : \underset{D}{err}\left(h_1\left(\gamma_2\left(S\right)\right)\right) > \eta\left(\varepsilon\right)\right) \le \left(n+1\right)^n \chi^l\left(\varepsilon\right)$$

$$\leq (n+1)^n \left(1 - x\frac{2-x}{(1-x)^2}\left(1 - \frac{\sigma^2}{\left(2\sqrt{n}-\mu\right)^2}\right)\right)^l$$

Bounding this with δ gives

$$D^l\left(S : \underset{D}{err}\left(h_1\left(\gamma_2\left(S\right)\right)\right) > \varepsilon\right) \leq (n+1)^n \left(1 - x\frac{2-x}{(1-x)^2}\left(1 - \frac{\sigma^2}{\left(2\sqrt{n}-\mu\right)^2}\right)\right)^l \leq \delta$$

Then, we take base 2 logarithm on both of the latter two sides to obtain

$$l \lg\left(1 - x\frac{2-x}{(1-x)^2}\left(1 - \frac{\sigma^2}{\left(2\sqrt{n}-\mu\right)^2}\right)\right) \leq \lg\delta - n\lg(n+1)$$

The minimum sample size is thus

$$l \geq \frac{\lg\delta - n\lg(n+1)}{\lg\left(1 - x\frac{2-x}{(1-x)^2}\left(1 - \frac{\sigma^2}{\left(2\sqrt{n}-\mu\right)^2}\right)\right)}.$$

Since $x\frac{2-x}{(1-x)^2}$, where $x = e^{-\beta/2}$, is a very small number, the resulting bound is very loose (as was the similar bound under the Dinur and Nissim framework discussed earlier). If n is small, the sample size l can be even greater than 2^n, which is the total number of all possible queries. With larger n, l becomes much smaller than 2^n. However, l is still a very large number. In order to reduce the sample size l, we need to find a more practical value instead of $x\frac{2-x}{(1-x)^2}$.

6.4.2 The Bound based on the Sample Size

Starting from Dinur and Nissim (2003), the sample size l is bounded by $n\lg^2 n$ if the fixed perturbation is less than $\sqrt{n}$. Therefore, we have a sufficient bound for the fixed-data perturbation case (see section 6.2 for the details):

$$l \geq \frac{\lg(\delta) - n\lg(n+1)}{\lg\left(1 - x\frac{2-x}{(1-x)^2}\right)} \geq n\lg^2 n$$

Consider the boundary case

$$n\lg^2 n = \frac{\lg(\delta) - n\lg(n+1)}{\lg(1-\xi(\varepsilon))}.$$

Then

$$\lg(1-\xi(\varepsilon)) = \frac{\lg(\delta) - n\lg(n+1)}{n\lg^2 n}$$

$$\xi(\varepsilon) = 1 - 2^{\frac{\lg(\delta) - n\lg(n+1)}{n\lg^2 n}}$$

Based on the above result for $\xi(\varepsilon)$, we replaced

$$x\frac{2-x}{(1-x)^2}$$

with

$$1 - 2^{\frac{\lg(\delta) - n\lg(n+1)}{n\lg^2 n}}$$

This formula provides a better value than $\xi(\varepsilon)$ while developing a tighter bound for the sample size in the variable-data perturbation case.

Since the reasoning used by Dinur and Nissim (2003) to arrive at $n\lg^2 n$ remains unchanged for our case, so we can use

$$1-2^{\frac{\lg(\delta)-n\lg(n+1)}{n\lg^2 n}}$$

in place of our $\eta(\varepsilon)$. This gives

$$(n+1)^n\left(1-\left(1-2^{\frac{\lg(\delta)-n\lg(n+1)}{n\lg^2 n}}\right)\left(1-\frac{\sigma^2}{\left(2\sqrt{n}-\mu\right)^2}\right)\right)^l \le \delta$$

from which we obtain

$$\lg(n+1)^n+\lg\left(1-\left(1-2^{\frac{\lg(\delta)-n\lg(n+1)}{n\lg^2 n}}\right)\left(1-\frac{\sigma^2}{\left(2\sqrt{n}-\mu\right)^2}\right)\right)^l \le \lg\delta$$

$$l\lg\left(1-\left(1-2^{\frac{\lg(\delta)-n\lg(n+1)}{n\lg^2 n}}\right)\left(1-\frac{\sigma^2}{\left(2\sqrt{n}-\mu\right)^2}\right)\right) \le \lg\delta-n\lg(n+1)$$

giving

$$l \ge \frac{\lg\delta-n\lg(n+1)}{\lg\left(1-\left(1-2^{\frac{\lg(\delta)-n\lg(n+1)}{n\lg^2 n}}\right)\left(1-\frac{\sigma^2}{\left(2\sqrt{n}-\mu\right)^2}\right)\right)} \qquad (6.7)$$

From formula (6.7) we can see that the sample size l decreases when μ and σ decrease.

6.4.3 Discussion

As we know from section 6.2, the larger the number of camouflage vectors s is, the larger the response intervals are, which lead to the larger perturbation mean and standard deviation. This result simply implies that sample size l increases with an increase of s.

Our experiments based on the three examples in Garfinkel et al. (2002) support these conclusions. The database has 14 records, $n = 14$. Three cases are considered in Table 6-2.

Table 6-2: The Relationship among μ, σ, s and l.

Network / Variable	$w = 3$ and $m = 1$	$w = 5$ and $m = 2$	$w = 7$ and $m = 3$
s	3	10	35
μ	2.0236	2.7760	3.3019
σ	1.1150	1.1114	1.174
l	213	217	223

From Table 6-2, we can see that the sample size l increases while μ, σ and s increase. These results of sample sizes are very close to the bound $n \lg^2 n$ from Dinur and Nissim (2003) and much less than $2^{14} = 16,384$.

6.5 Estimated the Mean and Standard Deviation

In the previous section, we derived a bound on the sample size, which is the minimum number of queries required to disclose the binary confidential information in a database protected by the variable-data perturbation method. The bound (see formula 6.7) is decided by four parameters: the number of database records n, the confidence parameter δ, and the mean μ and standard deviation σ of the perturbation distribution. Among these four parameters, n and δ are known and predetermined. In this section, we will develop a method to identify the estimated mean and standard deviation of the perturbation distribution.

Perturbations' mean μ and standard deviation σ are fixed in the Garfinkel et al. (2002) as soon as the algorithm design is finished, such as those networks for camouflage vectors in the CVC technique. However, the actual mean and standard deviation can be calculated only if all responses from 2^n queries are obtained, which is not practical in most situations. Instead of computing the true mean and standard deviation from 2^n queries, our heuristic method intends to estimates these two values approximately, denoted as $\tilde{\mu}$ and $\tilde{\sigma}$, by using the following random sampling method.

Let

i : index of query i

q_i : the i^{th} query

$\tilde{e}_{q_i}$: interval length of query q_i

$\tilde{\mu}_i$: mean of perturbations using queries $1,\cdots,i$

$\tilde{\sigma}_i$: standard deviation of perturbations using queries $1,\cdots,i$

l_i : sample size computed from $\tilde{\mu}_i$ and $\tilde{\sigma}_i$ using formula (6.7)

Table 6-3 lists the heuristic steps for estimating the mean, standard deviation and the bound on the sample size.

We use the network example in Garfinkel et al. (2002) to illustrate our heuristic. The basic setting for the network algorithm is: there are $n = 14$ database records, and parameters $w = 3$ and $m = 1$. The true mean and standard deviation computed from 2^{14} queries are $\mu = 2.023$ and $\sigma = 1.115$, which give a sample size $l = 213$ from formula (6.7). Also see Table 5-2 and Figure 5-1 for all camouflage vectors and the CVC network algorithm. Next, we show how the heuristic is applied to estimate $\tilde{\mu}$ and $\tilde{\sigma}$ for the CVC technique example in Garfinkel et al. (2002).

Table 6-3: Heuristic to Estimate the Mean $\tilde{\mu}$, Standard Deviation $\tilde{\sigma}$, and the Bound $\tilde{l}$.

```
Heuristic:
0.  for ( $i = 1; i \le 30; i{+}{+}$ )
        Generate query $q_i$ and record its perturbation $\tilde{e}_{q_i}$.
1. Generate query $q_i$ and record its perturbation $\tilde{e}_{q_i}$.
2. Compute $\tilde{\mu}_i$ and $\tilde{\sigma}_i$ using $\tilde{e}_{q_1}, \cdots, \tilde{e}_{q_i}$.
3. Compute $l_i$ from formula (6.7) using the estimated $\tilde{\mu}_i$
   and $\tilde{\sigma}_i$.
4. Increment $i$ and repeat step 1 to step 3 until $i \ge l_i$. This
   $l_i$ is the final bound on the sample size, $\tilde{l}$. $\tilde{\mu}_i$ and $\tilde{\sigma}_i$ are
   final values for the estimated $\tilde{\mu}$ and $\tilde{\sigma}$.
```

For example, the intruder sends a random query q_i to the database, asking how many employees in Company B have positive HIV (see Table 5-1). The query responds an interval answer as $[1,\ 2]$ (see Table 5-2 for the set of camouflage vectors), from which the random perturbation is recorded as $\tilde{e}_{q_i} = 2 - 1 = 1$. Continue sending queries and recording perturbations. The mean and standard deviation are computed as $\tilde{\mu}_i = \frac{\sum_{j=1}^{i} \tilde{e}_{q_j}}{i}$ and $\tilde{\sigma}_i = \sqrt{\frac{\sum_{j=1}^{i}\left(\tilde{e}_{q_j} - \tilde{u}_i\right)^2}{i}}$ using $\tilde{e}_{q_1}, \cdots, \tilde{e}_{q_i}$ when the number of queries is more than 30, which is considered a large enough or representative sample size in statistics. The bound on the sample size l_i is also computed using the estimated $\tilde{\mu}_i$ and $\tilde{\sigma}_i$ by formula (6.7). Keep updating the values of $\tilde{\mu}_i$, $\tilde{\sigma}_i$ and l_i while receiving new query responses. At the same time, i and l_i are compared. The intruder stops sending queries when $i \geq l_i$.

In Table 6-4, we simulate this heuristic by running programming language C++ and record the data for $\tilde{\mu}_i$, $\tilde{\sigma}_i$ and l_i while the number of queries increases until $i \geq l_i$.

Table 6-4: Summary of the Estimated $\tilde{\mu}_i$, $\tilde{\sigma}_i$ and l_i in the CVC Example Network.

i^{th} Query	$\tilde{\mu}_i$	$\tilde{\sigma}_i$	l_i
30	1.935	0.948	210
50	2.118	0.983	211
80	2.098	1.018	211
110	2.099	1.036	212
140	2.135	1.069	212
170	2.140	1.085	213
200	2.124	1.102	213
212	2.118	1.105	213

From our example, the sample size computed from the estimated $\tilde{\mu}$ and $\tilde{\sigma}$ is the same as the true bound, $\tilde{l} = l = 213$. Although $\tilde{\mu}$ and $\tilde{\sigma}$ are not exactly equivalent to the true μ and σ, they are close enough to get the same or very similar bound on the sample size. After computing the bound, we run the LP algorithm (specified in section 6.2) to discover the true confidential vector from l perturbed answers.

In this chapter, we build PAC models and derive the bounds for both the fixed data perturbation and the variable data perturbation methods. First, according to Dinur and

Nissim (2003)'s Disqualifying Lemma, we derive a PAC bound for the fixed data perturbation, which is

$$l \geq \frac{\lg(\delta) - n\lg(n+1)}{\lg(1-\xi(\varepsilon))}$$

This bound is much looser than the sampling bound $n\lg^2 n$ developed in Dinur and Nissim (2003). The second bound is derived from our Lemma 2 for the variable data perturbation, which is

$$l \geq \frac{\lg\delta - n\lg(n+1)}{\lg\left(1 - x\frac{2-x}{(1-x)^2}\left(1 - \frac{\sigma^2}{\left(2\sqrt{n}-\mu\right)^2}\right)\right)}$$

where $x = e^{-\beta/2}$. This bound is still not tight enough to be useful. Then we have the third bound

$$l \geq \frac{\lg\delta - n\lg(n+1)}{\lg\left(1 - \left(1 - 2^{\frac{\lg(\delta)-n\lg(n+1)}{n\lg^2 n}}\right)\left(1 - \frac{\sigma^2}{\left(2\sqrt{n}-\mu\right)^2}\right)\right)}$$

This bound is very practical and can be applied in a real database. In the next chapter, we will design some experiments to test our bound under different situations.

CHAPTER 7
EXPERIMENTAL DESIGN AND RESULTS

In Chapter 6, we built the PAC model and derived an error bound from the new Disqualifying Lemma (Lemma 2) for the variable data perturbation method. The bound determines the number of queries necessary to compromise binary confidential data in a database. In this chapter, we create a simulated database with one binary confidential field which is protected by the variable data perturbation. All experiments are conducted to illustrate our results from the previous chapter. Computational results are analyzed and compared to examine how the perturbations' mean, standard deviation and distribution affect the bound on sample size and the level of disclosure accuracy.

7.1 Experimental Environment and Setup

Experiments are designed to empirically illustrate the level of accuracy with which an adversary can compromise a database protected by the variable-data perturbation method within a specific number of queries derived from our new Disqualifying Lemma (Lemma 2).

Due to the limited capacity of our testing software CPLEX 8.0 (ILOG), our experiments can only consider the bound in formula (6.7) because the sample size computed from formula (6.6) generates an LP that is too large to be solved. For the same reason, we have to relax the requirement that n needs to be large enough for formula (6.7) to hold for a specified ε and instead choose a relatively small n for solving the LP problem. Thus, all bounds used in our tests are computed from formula (6.7). This bound is sufficient for large enough n so we are examining that would apply to more benign distribution cases. As we see, the cases investigated seem to show some efficacy even under the various cases studied here.

In our tests, the simulated database has 100 records with one confidential binary field. For each query we sample a perturbation width $\tilde{e}_q$ and generate an interval answer by randomly splitting $\tilde{e}_q$ into two values, each of which is deducted from or added to the true query answer to construct the lower and upper bound. The heuristic in Table 6-3 shows how to estimate the mean and standard deviation of the perturbations from which the bound on the sample size l can be computed. The LP discussed in Chapter 6 is applied by the adversary to output the candidate binary vector. The sample size is computed using (6.7):

$$l \geq \frac{\lg \delta - n \lg\left(n+1\right)}{\lg\left(1-\left(1-2^{\frac{\lg(\delta)-n\lg(n+1)}{n\lg^2 n}}\right)\left(1-\frac{\sigma^2}{\left(2\sqrt{n}-\mu\right)^2}\right)\right)}$$

where $n = 100$ and $\delta = 0.05$ in our experiments.

Four types of discrete perturbation distributions for s, $e_q \sim D\left(\mu, \sigma^2\right)$, are considered in the experiments:

(1) Uniform distribution

(2) Symmetric Distribution

(3) Distribution with Positive Skewness (Skew to the right)

(4) Distribution with Negative Skewness (Skew to the left)

There are four cases with different means and standard deviations under each type of distribution. So, a total of 16 experiments are conducted. See Table 7-1 for the summary of four cases.

Table 7-1: Summary of Four cases with Different Means and Standard Deviations.

Variables / Cases	μ	σ
Case 1	high	high
Case 2	high	low
Case 3	low	high
Case 4	low	low

7.2 Data Generation

During the experiments, we assume all perturbations are distributed within different intervals $\left[a,b\right]$ under each of the four cases. Table 7-2 lists those four intervals.

Table 7-2: The Intervals of $\left[a,b\right]$ under the Four Cases.

Variables / Cases	a	b
Case 1: high μ, high σ	1	18
Case 2: high μ, low σ	5	14
Case 3: low μ, high σ	1	10
Case 4: low μ, low σ	3	7

In each test, we use the inverse transform method to sample random perturbations from a given distribution. The LP algorithm takes those perturbation values of e_q as the inputs and then outputs a candidate binary vector. Then we can compute the errors, which record the difference between the candidate vector and the true confidential data. We define an error rate as the percentage of errors that are present in the total number of database records. The average error rate is computed by running each case 100 times to reduce possible bias. The mean and standard deviation of every perturbation distribution need to satisfy three requirements (assumptions discussed in section 6.3 for Lemma 2 proof): (1) $(\mu+\sigma)/2<\sqrt{n}$; (2) $\mu>\sigma$; and (3) $\mu<\sqrt{n}$. In our tests, $\sqrt{n}=10$, and different means and standard deviations are summarized in Table 7-3.

The following part presents 16 distribution plots with different means, standard deviations and distribution types. All perturbations randomly generated from those distributions for the tests are shown in the Appendix. All experiments results are summarized in Table 7-3.

1. Uniform Distribution

In this category, perturbations are randomly generated from the following given uniform distributions. Every perturbation value can be produced with the same probability. Four different distributions with different means and standard deviations are shown in Figure 7-1.

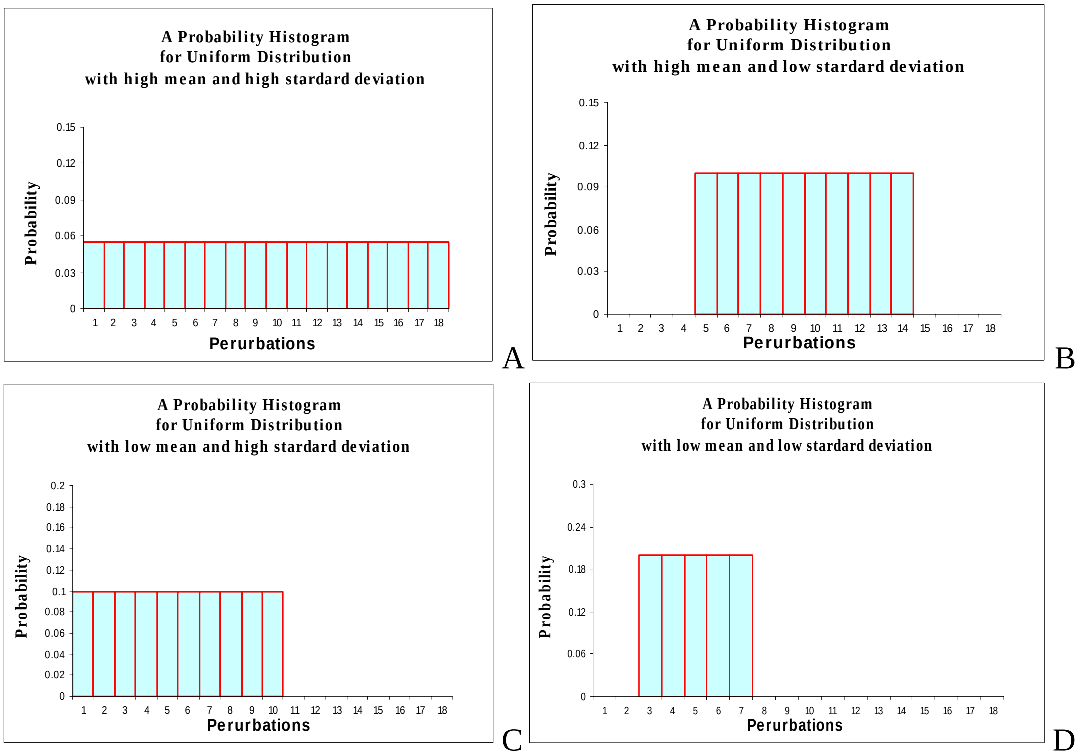


Figure 7-1: Plots of Four Uniform Distributions of Perturbations at Different Means and Standard Deviations. A) Case 1: high μ, high σ, B) Case 2: high μ, low σ, C) Case 3: low μ, high σ and D) Case 4: low μ, low σ.

2. Symmetric Distribution

Perturbations are distributed symmetrically in the following four cases. Every distribution’s mean, median and mode are all equal. The four given distributions used to generate random perturbations in our tests are shown in Figure 7-2.

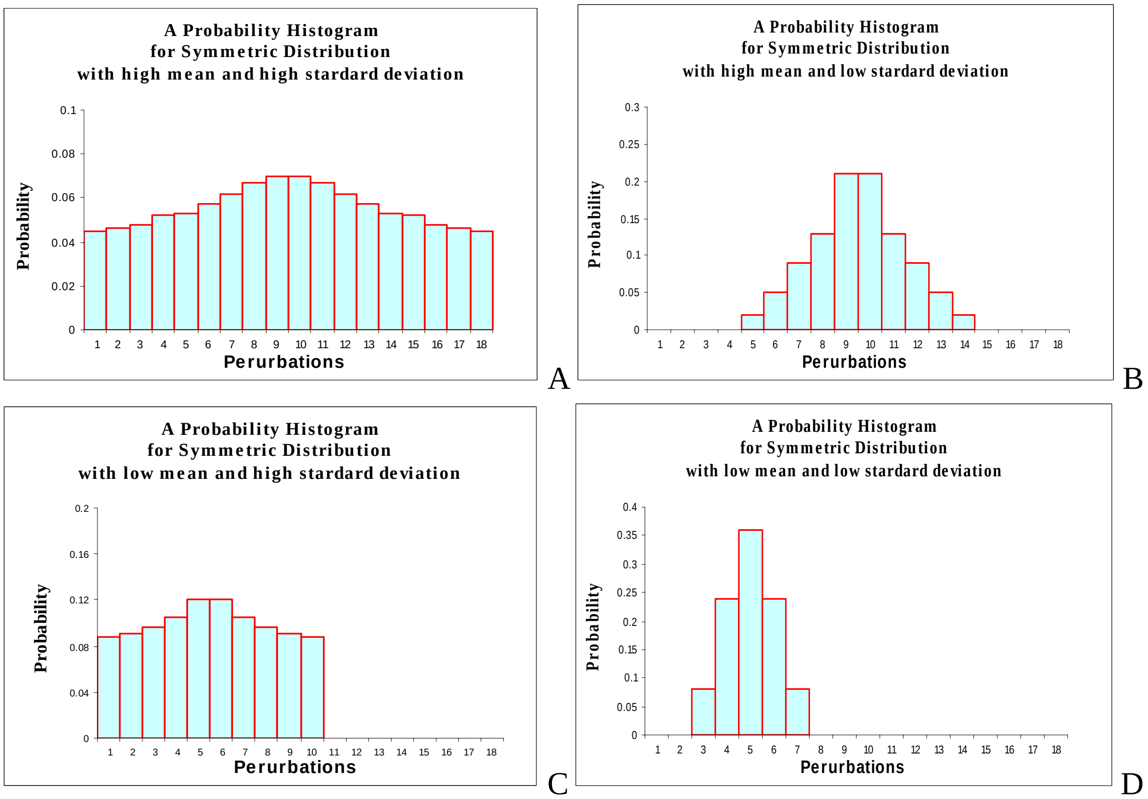


Figure 7-2: Plots of Four Symmetric Distributions of Perturbations at Different Means and Standard Deviations. A) Case 1: high μ, high σ, B) Case 2: high μ, low σ, C) Case 3: low μ, high σ and D) Case 4: low μ, low σ.

3. Distribution with Positive Skewness

Positive skewness indicates that the distribution skews to the right and its mean is greater than the median. Most of the perturbations are less than the average. Random perturbations are generated from the following four given distributions with different means and standard deviations shown in Figure 7-3.

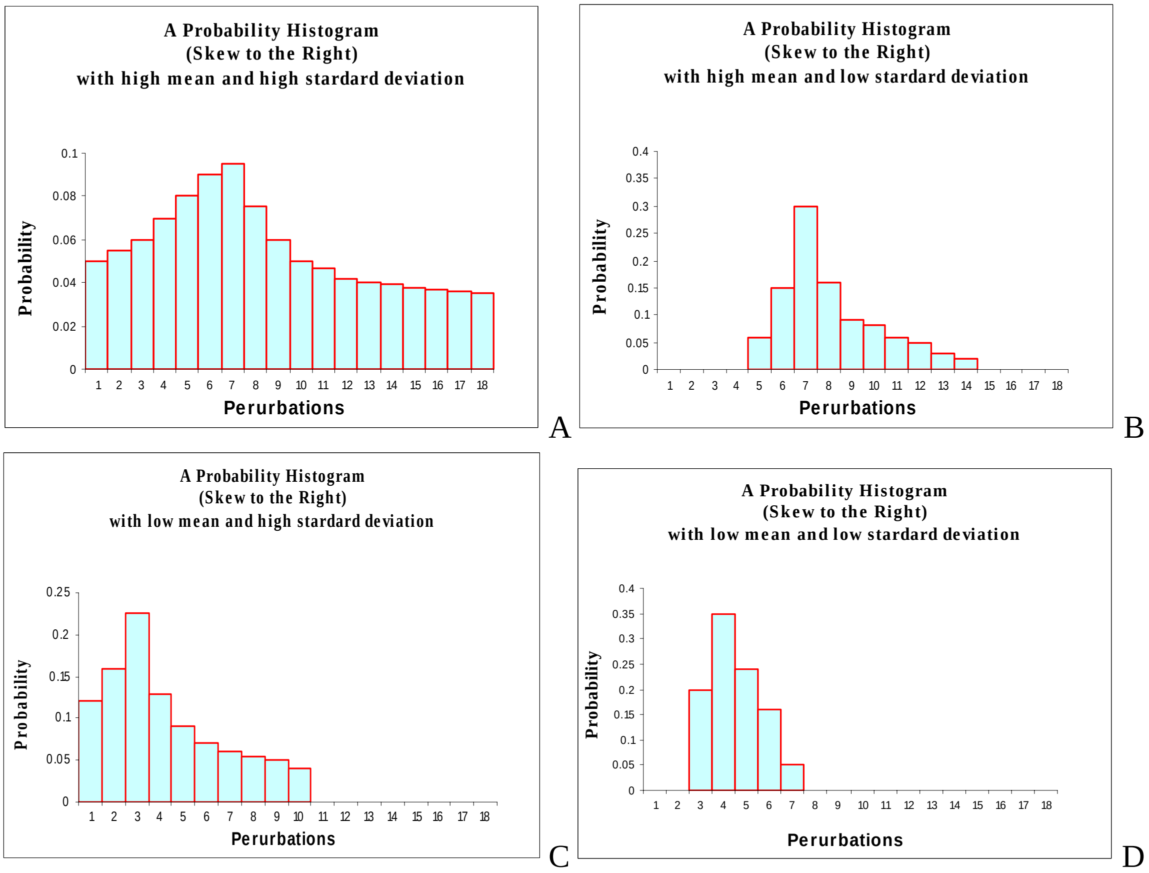


Figure 7-3: Plots of Four Distributions with Positive Skewness of Perturbations at Different Means and Standard Deviations. A) Case 1: high μ, high σ, B) Case 2: high μ, low σ, C) Case 3: low μ, high σ and D) Case 4: low μ, low σ.

4. Distribution with Negative Skewness

Negative skewness indicates the distribution skews to the left and its median is greater than the mean which is less than most of the values. Random perturbations are generated from the following four given distributions with different means and standard deviations shown in Figure 7-4.

7.3 Experimental Results

Experiments are conducted to disclose the confidential binary data in a simulated database. Four types of perturbation distributions, each of which has four cases, are considered. The LP algorithm outputs the candidate confidential vector at the end by running C++ and CPLEX. Our program, which simulates queries and their perturbations, does not deal with the case where the same query should be modified by the same perturbation, since the probability that one query is chosen twice during one run is very small ($\frac{1}{2^{100}}$).

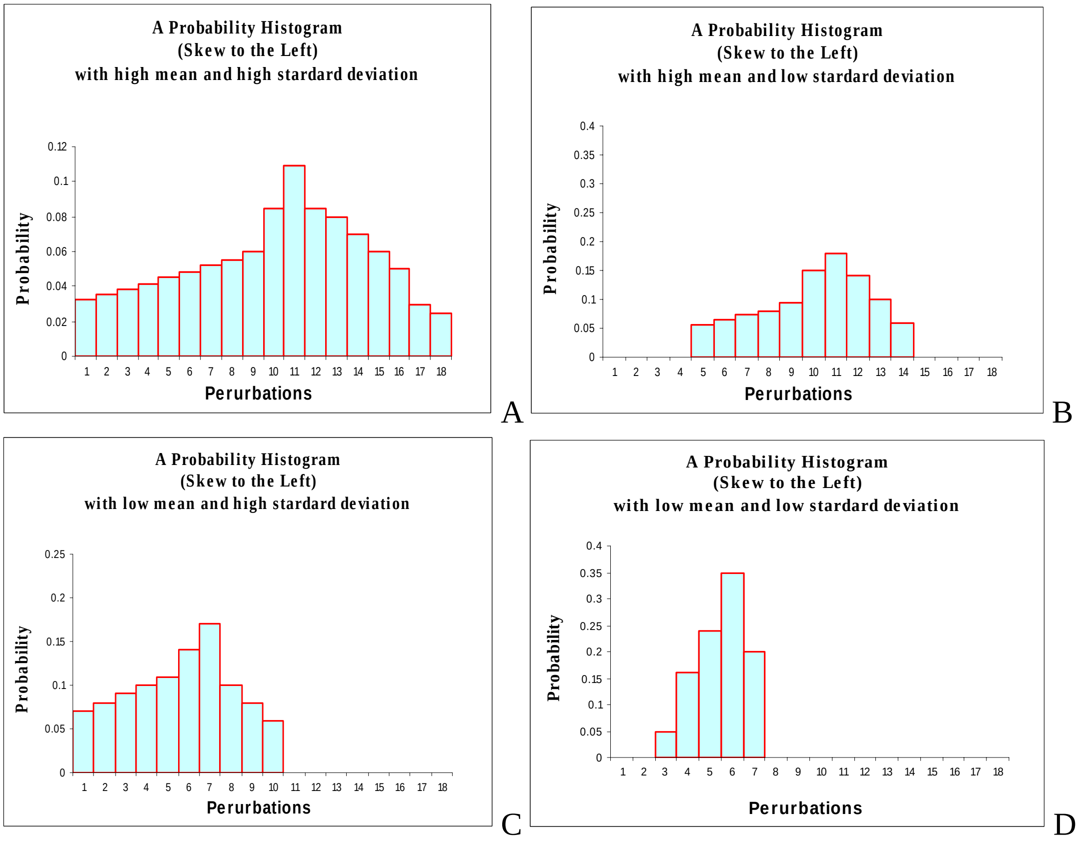


Figure 7-4: Plots of Four Distributions with Positive Skewness of Perturbations at Different Means and Standard Deviations. A) Case 1: high μ, high σ, B) Case 2: high μ, low σ, C) Case 3: low μ, high σ and D) Case 4: low μ, low σ.

7.3.1 Experiment 1

The bound on the sample size (from formula 6.7) and average error rate are computed. Table 7-3 lists the information about the mean and standard deviation, and also records computational results about the sample size and average error rate.

Table 7-3: Experiments Results on 16 tests with the Means, Standard Deviations, Sample Sizes and Average Error Rates.

Variables / Distributions and Cases		μ	σ	$(\mu+\sigma)/2$	l	Average Error Rate (%)
Uniform	Case 1	9.50	4.91	7.20	5715	12.12
	Case 2	9.50	2.60	6.05	4719	14.05
	Case 3	5.50	2.60	4.05	4569	13.29
	Case 4	5.50	0.87	3.18	4443	13.83
Symmetric	Case 1	9.50	4.85	7.18	5678	12.28
	Case 2	9.50	1.96	5.73	4584	13.94
	Case 3	5.50	2.74	4.12	4587	13.71
	Case 4	5.00	1.06	3.03	4438	13.84
Positive Skewness	Case 1	8.36	4.74	6.56	5342	13.15
	Case 2	8.12	2.16	5.14	4574	14.26
	Case 3	4.22	2.53	3.37	4537	13.75
	Case 4	4.51	1.13	2.82	4440	13.46
Negative Skewness	Case 1	9.99	4.41	7.20	5536	12.70
	Case 2	9.99	2.47	6.23	4717	13.79
	Case 3	5.60	2.54	4.07	4564	12.83
	Case 4	5.49	1.13	3.31	4443	13.49

Figure 7-5 shows that case 1 can always be compromised more than the other three cases, no matter what type of distribution it has. Although the result seems counterintuitive at first sight, it does support one of our assumptions in section 6.3. A high mean and high standard deviation of a perturbation distribution indicate that many query responses have large perturbations which may perturb the true answer too much to be useful for the user, while other queries provide very tight answers which can reveal the confidential data easily. It explains why case 1 with a high mean and high standard deviation still can have a low error rate.

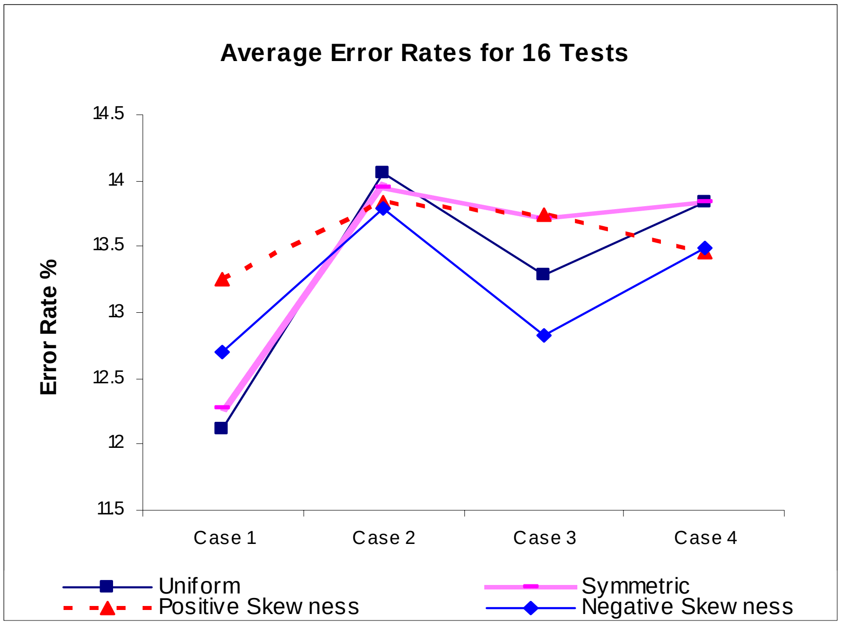


Figure 7-5: Plot of Average Error Rates (%) for 16 Tests.

Among the four cases, case 2 is more difficult to disclose for any type of distribution. This is true because high mean and low standard deviation indicate that most of the perturbations are clustered around the high average mean value which provides good protection to the database, but the user may get little information from the query answers.

An example similar to case 2 occurred in Garfinkel et al. (2002). The CVC technique designed three sample networks to construct the camouflage vectors for an example database (Table 5-1). Among those three networks, the one with the perfect column balancing, which provides the best protection to the database according to the paper, has a high perturbation mean $\mu = 3.302$ which is close to $\sqrt{n} = 3.742$ and low standard deviation $\sigma = 1.174$, similar to our case 2. Based on our experimental results, this network does protect the database well (recall that case 2 always has a high error rate), however, most of the time, the user get little information from query answers protected by this security method. Figure 7-6 shows 61% of the perturbations are clustered around the mean, and the standard deviation is small.

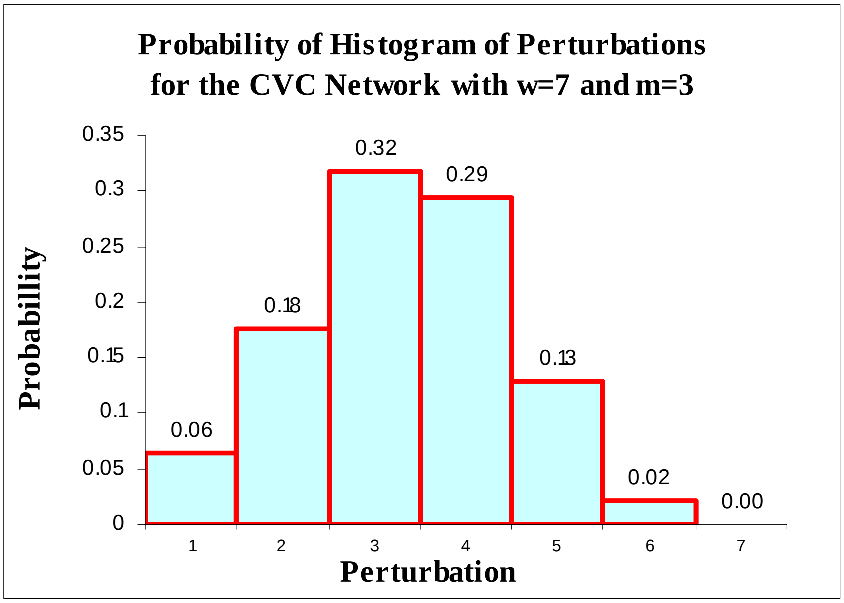


Figure 7-6: The Probability Histogram of Perturbation Distribution for the CVC Network

Error rates of case 3 and case 4 are always between those of case 1 and case 2, and they offer the user more accurate data than case 2. Case 3 is supposed to have a lower error rate than case 1 because of its low mean and high standard deviation which indicate most of the query answers have small perturbations. However, a low mean and high standard also generate a smaller sample size which may explain why case 3 has a higher error rate than case 1 in our tests.

Figure 7-7 records the bounds on the sample size for 16 tests. It shows that the bound increases with increases of the mean and standard deviation in all types of distributions. Dinur and Nissim (2003) gave the bound for the fixed data perturbation as $n \lg^2 n$, which is 4,415 in our experiments. Most of our bounds are a little looser than this value.

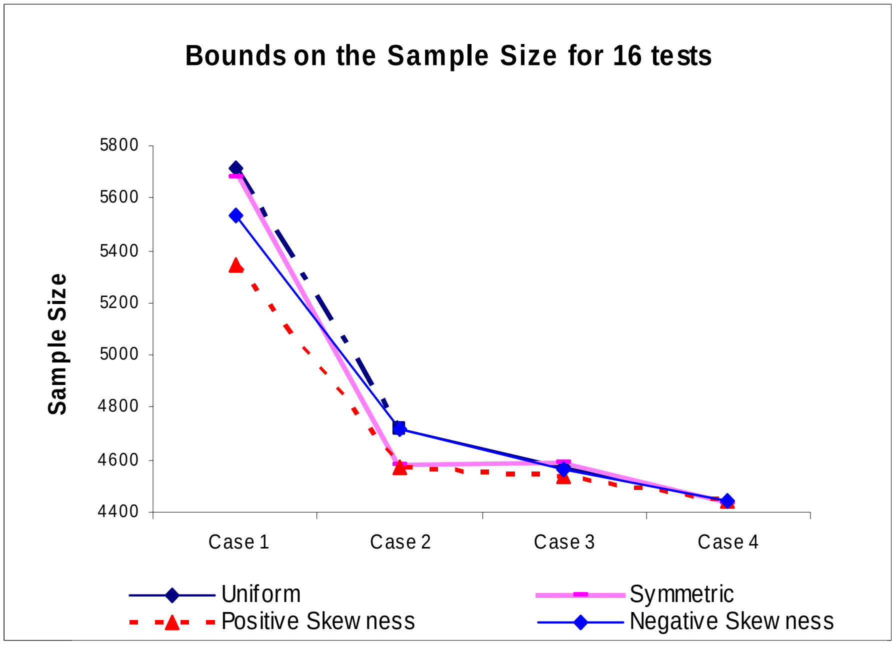


Figure 7-7: Plot of Bounds on the Sample Size for 16 Tests.

7.3.2 Experiment 2

Results in Experiment 1 are based on the sample sizes computed from different means and standard deviations in 16 cases. In order to reduce the bias because of the different sample sizes, we also computed the average error rates by using $l = 6,000$ for each case. Table 7-4 shows the average error rate for each case when the sample size is the same. All other variables comply with those in Table 7-3.

Table 7-4: Experimental Results on the Average Error Rates with $l = 6,000$ for 16 Cases.

Variables Distributions and Cases		μ	σ	Average Error Rate (%) $l = 6,000$
Uniform	Case 1	9.50	4.91	11.91
	Case 2	9.50	2.60	12.92
	Case 3	5.50	2.60	11.72
	Case 4	5.50	0.87	12.28

Table 7-4. Continued.

Variables Distributions and Cases		μ	σ	Average Error Rate (%) $l = 6,000$
Symmetric	Case 1	9.50	4.85	12.16
	Case 2	9.50	1.96	12.98
	Case 3	5.50	2.74	11.51
	Case 4	5.00	1.06	12.21
Positive Skewness	Case 1	8.36	4.74	12.11
	Case 2	8.12	2.16	12.52
	Case 3	4.22	2.53	11.35
	Case 4	4.51	1.13	12.28
Negative Skewness	Case 1	9.99	4.41	12.09
	Case 2	9.99	2.47	12.53
	Case 3	5.60	2.54	11.24
	Case 4	5.49	1.13	12.25

As we suspected case 3, with low mean and high standard deviation, becomes the most unsafe situation for the database security conflicting with the conclusions from Table 7-3. Figure 7-8 displays the results for 16 tests.

In sum, experiment 1 suggests that case 1 is always worse than case 2 in terms of protecting the database. We can conclude from the test results from experiment 1 that a database may be compromised more easily if its perturbation distribution has a high mean and high standard deviation. A high mean and low standard deviation can best protect a database, but the query answer may be useless because of the large perturbations. Case 3, with low mean and high standard deviation, usually provides the user with the most useful query responses. Experiment 2 shows that with the same sample sizes, a database with perturbations in case 3 is the easiest to be discovered.

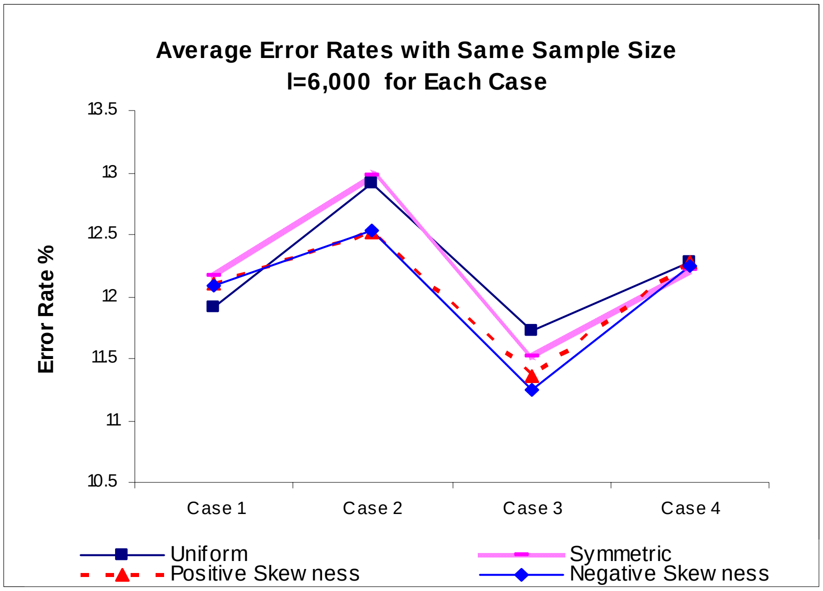


Figure 7-8: Plot of Average Error Rates (%) for 16 tests with the Same Sample Size $l = 6,000$.

In general, we see that a high level of protection may yield answers that are not useful and useful answers compromise the database. The experimental results also support our observation from chapter 6 and show that with high probability, the binary confidential information in a database protected by the variable data perturbation can be disclosed at small error within a certain number of queries as suggested by Inequality (6.7).

CHAPTER 8
CONCLUSION

8.1 Overview and Contribution

In this dissertation, we address the statistical database security problems from a new perspective by applying PAC learning theory. By learning from examples, the main idea of the PAC model is that the hypothesis generated from the learning algorithm approximates the target concept with a high probability at small error in polynomial time and/or space. By deploying the PAC learning theory, we regard the adversary of the database as a learner who tries to discover the confidential data within a certain number of queries. This new approach is different from the traditional methods in the literature. Instead of building models to protect the confidential information, we focus on how to compromise the database, therefore finding out how much protection is necessary to prevent the disclosure of sensitive information contained in a database.

First, we review the SDC methods in the literature and focus our research on a new data perturbation method. Inspired by the CVC interval protection technique developed by Garfinkel et al. (2002), we define this new technique as the variable data perturbation method which can be viewed as modifying the confidential information by adding discrete noise e_q. Although the random perturbations have an unknown distribution from an intruder's perspective, we can estimate the parameters, such as its mean and variance using a heuristic method detailed in Chapter 6.

We also extend the work by Dinur and Nissim (2003). In their study, all queries have fixed perturbations e. No information is provided about the distribution of the perturbations. They derived a bound on the sample size, within which the true confidential binary string should be discovered with high probability by running an LP algorithm. It is assumed that the fixed perturbations may happen on either side of the query responses while setting up constraints for the LP algorithm.

In our paper, we interpret their results within the methodology of PAC learning theory and derive a bound for the fixed data perturbation method. Then, we develop the PAC bounds on the sample size from our Lemma 2 for the variable data perturbation method. Within the PAC number of queries, a database protected by the variable data perturbation can be compromised with a high probability at small error. Since the bound is decided by parameters, such as the mean and standard deviation, a heuristic method is also introduced to estimate these two values.

To illustrate our results, we perform a number of numerical experiments conducted on a simulated database over four types of perturbation distributions with different means and standard deviations. The test results show that these databases can be compromised at fairly high levels and also show that the mean and standard deviation of the perturbation distribution are more important factors than the type of the distribution in terms of affecting the error rate and the sample size.

8.2 Limitations

There are three main limitations in our work on the database security problems.

First, we only consider the case that the confidential data is perturbed by discrete random perturbations even though continuous noises can also be added to the database protected by the variable data perturbation method. Second, when deriving the bound, we assume that the confidential item is binary valued . In general, a confidential field can contain many types of data, such as real numbers or categorical data. So, this assumption may constrain the application of this bound. Last, the experiments are conducted on a simulated database rather than a real database. Moreover, the simulated database used in the experiments had a relatively small number of records (100) due to the limitations of our testing software.

8.3 Directions for Future Research

In future research, we can consider other types of confidential items such as real-valued or categorical. We may also examine other types of perturbations, such as real-valued ones. Even within Dinur and Nissim (2003) type of setting we might consider a case where their perturbation is fixed but initially drawn from some known distribution.

A typical example for the variable data perturbation is the CVC technique developed by Garfinkel et al. (2002). We simulated their network algorithm with different parameters w and m on the example database in Garfinkel et al. (2002) and observed on a number of cases that, given a large-enough number of random queries, all camouflage vectors could be discovered by running the LP algorithm used in Chapter 6. Based on these experimental results, we conjecture that every camouflage vector in the Bin-CVC technique is an extreme point of a polyhedron formed by all the 2^n queries and, conversely, that all the extreme points are camouflage vectors. How this (if true) pertains to polytopes formed with a subset of the 2^n possible queries needs to be investigated. We suspect that if the output from the LP algorithm is an integer vector, then it will be one of the extreme points, therefore, one of the camouflage vectors. This is an important possible weakness of the CVC method since there are generally few camouflage vectors and one is the true vector of database values. The discovery of the camouflage vectors reduces the intrusion problem to discovering which amoung a small number of vectors, is the true vector. Insider information on small number of query values could easily determine which is the true vector.

Muralidhar et al. (2004) compromised CVC interval protection empirically by employing a simple deterministic procedure. They also claimed that if the CVC technique intends to prevent the interval disclosure, such as increasing the number of camouflage vectors, data utility has to be damaged substantially. Our future research will try to extend their work and propose a more general theoretical method to address the problem.

Since choosing an appropriate security method depends greatly on how well it can balance the tradeoff between information loss and disclosure risk, our future task is to develop a general performance measurement which can be used to assess comprehensively the disclosure risk and information loss for the variable-data perturbation method, such as the measure for the interval protection. By applying this measure, we would be able to check the utility of the interval answer from the Bin-CVC technique and investigate whether CVC interval protection is practical or whether the quality of responses to queries outweighs the high level of protection for the database. We hope this evaluation scheme can become a guideline for selecting ideal security methods in SDBs under some specific situations.

APPENDIX A
NOTATION TABLES

Table A-1: Notations in Machine Learning and PAC Learning Theory.

Notation	Definition
f	Target concept (or target function)
x	Instance
X	Instance space or Input space
y	Output
Y	Output space
S	Sample
l	Sample size
n	Number of attributes or number of records in the database
h	Hypothesis
H	Hypothesis space
$\|H\|$	Cardinality of H
C	Concept space
D	Probability distribution
$\underset{D}{err}(\ )$	Probability of error
ε	Accuracy parameter
δ	Confidence parameter
E_i	Event i
ε_s	Training error
d	VC dimension
$L(\ )$	Loss function
$R(\alpha)$	Risk functional
z	Observation pairs
$g(z,\alpha)$	Set of target functions with parameters $\alpha \in \Lambda$
$F(z)$	Unknown Probability distribution
$R_{emp}(\ )$	Empirical risk
$R_{struct}(\ )$	Structural risk
$R_{bound}(\ )$	Risk bound

Table A-2: Notations in Statistical Disclosure Control Methods

Notation	Definition
μ	Mean
σ^2	Variance
d	True confidential vector
e	Perturbation vector
π	Random variable with a normal distribution
V	Covariance matrix
τ	Number of camouflage vectors
P	Set of camouflage vectors
P^j	j^{th} camouflage vector, $j=1,\cdots,k$
p_i^j	i^{th} element in j^{th} camouflage vector, $i=1,\cdots,n$
q	Query
$u(\ )$	Upper bound
$l(\ )$	Lower bound
$I(\ )$	Interval between $l(\ )$ and $u(\ )$
w	Total number of paths in the network algorithm
m	Number of paths consisting only of true value edges
$p*$	Proportion of ones in the confidential vector
p^j	Proportion of ones in the j^{th} camouflage vector
$card(q)$	Cardinality of query q
e_q	Perturbation vector generated from an algorithm
$a(q)$	True query answer
$A(q)$	Perturbed query answer
k	Precision parameter
K	Set of precision parameters
$dist(\ )$	Hamming distance

APPENDIX B
DATA GENERATED FOR THE UNIFORM DISTRIBUTION

Table B-1: Case 1 with High Mean and High Standard Deviation

Perturbation	**Frequency**	**Probability**	**CDF**
1	600	0.06	0.06
2	600	0.06	0.11
3	600	0.06	0.17
4	600	0.06	0.22
5	600	0.06	0.28
6	600	0.06	0.33
7	600	0.06	0.39
8	600	0.06	0.44
9	600	0.06	0.50
10	600	0.06	0.56
11	600	0.06	0.61
12	600	0.06	0.67
13	600	0.06	0.72
14	600	0.06	0.78
15	600	0.06	0.83
16	600	0.06	0.89
17	600	0.06	0.94
18	600	0.06	1.00
Total	10800	1.00	

Table B-2: Case 2 with High Mean and Low Standard Deviation

Perturbation	**Frequency**	**Probability**	**CDF**
1	0	0	0
2	0	0	0
3	0	0	0
4	0	0	0
5	1000	0.1	0.1
6	1000	0.1	0.2
7	1000	0.1	0.3
8	1000	0.1	0.4
9	1000	0.1	0.5
10	1000	0.1	0.6
11	1000	0.1	0.7
12	1000	0.1	0.8
13	1000	0.1	0.9
14	1000	0.1	1

Table B-2. Continued

Perturbation	**Frequency**	**Probability**	**CDF**
15	0	0	
16	0	0	
17	0	0	
18	0	0	
Total	10000	1	

Table B-3: Case 3 with Low Mean and High Standard Deviation

Perturbation	**Frequency**	**Probability**	**CDF**
1	1000	0.1	0.1
2	1000	0.1	0.2
3	1000	0.1	0.3
4	1000	0.1	0.4
5	1000	0.1	0.5
6	1000	0.1	0.6
7	1000	0.1	0.7
8	1000	0.1	0.8
9	1000	0.1	0.9
10	1000	0.1	1
11	0	0	
12	0	0	
13	0	0	
14	0	0	
15	0	0	
16	0	0	
17	0	0	
18	0	0	
Total	10000	1	

Table B-4: Case 4 with Low Mean and Low Standard Deviation

Perturbation	**Frequency**	**Probability**	**CDF**
1	0	0	0
2	0	0	0
3	2000	0.2	0.2
4	2000	0.2	0.4
5	2000	0.2	0.6
6	2000	0.2	0.8
7	2000	0.2	1
8	0	0	
9	0	0	
10	0	0	
11	0	0	
12	0	0	
13	0	0	

Table B-4 Continued

Perturbation	**Frequency**	**Probability**	**CDF**
15	0	0	
14	0	0	
16	0	0	
17	0	0	
18	0	0	
Total	10000	1	

APPENDIX C
DATA GENERATED FOR THE SYMMETRIC DISTRIBUTION

Table C-1: Case 1 with High Mean and High Standard Deviation

Perturbation	Frequency	Probability	CDF
1	450	0.045	0.045
2	460	0.046	0.091
3	480	0.048	0.139
4	520	0.052	0.191
5	530	0.053	0.244
6	570	0.057	0.301
7	620	0.062	0.363
8	670	0.067	0.43
9	700	0.07	0.5
10	700	0.07	0.57
11	670	0.067	0.637
12	620	0.062	0.699
13	570	0.057	0.756
14	530	0.053	0.809
15	520	0.052	0.861
16	480	0.048	0.909
17	460	0.046	0.955
18	450	0.045	1
Total	10000	1	

Table C-2: Case 2 with High Mean and Low Standard Deviation

Perturbation	Frequency	Probability	CDF
1	0	0	0
2	0	0	0
3	0	0	0
4	0	0	0
5	200	0.02	0.02
6	500	0.05	0.07
7	900	0.09	0.16
8	1300	0.13	0.29
9	2100	0.21	0.5
10	2100	0.21	0.71
11	1300	0.13	0.84
12	900	0.09	0.93
13	500	0.05	0.98
14	200	0.02	1

Table C-2 Continued

Perturbation	Frequency	Probability	CDF
15	0	0	
16	0	0	
17	0	0	
18	0	0	
Total	10000	1	

Table C-3: Case 3 with Low Mean and High Standard Deviation

Perturbation	Frequency	Probability	CDF
1	875	0.0875	0.0875
2	905	0.0905	0.178
3	970	0.097	0.275
4	1050	0.105	0.38
5	1200	0.12	0.5
6	1200	0.12	0.62
7	1050	0.105	0.725
8	970	0.097	0.822
9	905	0.0905	0.9125
10	875	0.0875	1
11	0	0	
12	0	0	
13	0	0	
14	0	0	
15	0	0	
16	0	0	
17	0	0	
18	0	0	
Total	10000	1	

Table C-4: Case 4 with Low Mean and Low Standard Deviation

Perturbation	Frequency	Probability	CDF
1	0	0	0
2	0	0	0
3	800	0.08	0.08
4	2400	0.24	0.32
5	3600	0.36	0.68
6	2400	0.24	0.92
7	800	0.08	1
8	0	0	
9	0	0	
10	0	0	
11	0	0	
12	0	0	
13	0	0	

Table C-4. Continued.

Perturbation	**Frequency**	**Probability**	**CDF**
14	0	0	
14	0	0	
15	0	0	
16	0	0	
17	0	0	
18	0	0	
Total	10000	1	

APPENDIX D
DATA GENERATED FOR THE DISTRIBUTION WITH POSITIVE SKEWNESS

Table D-1: Case 1 with High Mean and High Standard Deviation

Perturbation	**Frequency**	**Probability**	**CDF**
1	500	0.05	0.05
2	550	0.055	0.105
3	600	0.06	0.165
4	700	0.07	0.235
5	800	0.08	0.315
6	905	0.0905	0.4055
7	950	0.095	0.5005
8	755	0.0755	0.576
9	600	0.06	0.636
10	500	0.05	0.686
11	470	0.047	0.733
12	420	0.042	0.775
13	400	0.04	0.815
14	390	0.039	0.854
15	380	0.038	0.892
16	370	0.037	0.929
17	360	0.036	0.965
18	350	0.035	1
Total	10000	1	

Table D-2: Case 2 with High Mean and Low Standard Deviation

Perturbation	**Frequency**	**Probability**	**CDF**
1	0	0	0
2	0	0	0
3	0	0	0
4	0	0	0
5	600	0.06	0.06
6	1500	0.15	0.21
7	3000	0.3	0.51
8	1600	0.16	0.67
9	900	0.09	0.76
10	800	0.08	0.84
11	600	0.06	0.9
12	500	0.05	0.95
13	300	0.03	0.98
14	200	0.02	1

Table D-2. Continued.

Perturbation	Frequency	Probability	CDF
15	0	0	
16	0	0	
17	0	0	
18	0	0	
Total	10000	1	

Table D-3: Case 3 with Low Mean and High Standard Deviation

Perturbation	Frequency	Probability	CDF
1	1200	0.12	0.12
2	1600	0.16	0.28
3	2250	0.225	0.505
4	1300	0.13	0.635
5	900	0.09	0.725
6	700	0.07	0.795
7	600	0.06	0.855
8	550	0.055	0.91
9	500	0.05	0.96
10	400	0.04	1
11	0	0	
12	0	0	
13	0	0	
14	0	0	
15	0	0	
16	0	0	
17	0	0	
18	0	0	
Total	10000	1	

Table D-4: Case 4 with Low Mean and Low Standard Deviation

Perturbation	Frequency	Probability	CDF
1	0	0	0
2	0	0	0
3	2000	0.2	0.2
4	3500	0.35	0.55
5	2400	0.24	0.79
6	1600	0.16	0.95
7	500	0.05	1
8	0	0	
9	0	0	
10	0	0	
11	0	0	
12	0	0	
13	0	0	

Table D-4. Continued.

Perturbation	**Frequency**	**Probability**	**CDF**
14	0	0	
15	0	0	
16	0	0	
17	0	0	
18	0	0	
Total	10000	1	

APPENDIX E
DATA GENERATED FOR THE DISTRIBUTION WITH NEGATIVE SKEWNESS

Table E-1: Case 1 with High Mean and High Standard Deviation

Perturbation	Frequency	Probability	CDF
1	320	0.032	0.032
2	350	0.035	0.067
3	380	0.038	0.105
4	410	0.041	0.146
5	450	0.045	0.191
6	480	0.048	0.239
7	520	0.052	0.291
8	550	0.055	0.346
9	600	0.06	0.406
10	850	0.085	0.491
11	1090	0.109	0.6
12	850	0.085	0.685
13	800	0.08	0.765
14	700	0.07	0.835
15	600	0.06	0.895
16	500	0.05	0.945
17	300	0.03	0.975
18	250	0.025	1
Total	10000	1	

Table E-2: Case 2 with High Mean and Low Standard Deviation

Perturbation	Frequency	Probability	CDF
1	0	0	0
2	0	0	0
3	0	0	0
4	0	0	0
5	550	0.055	0.055
6	650	0.065	0.12
7	750	0.075	0.195
8	800	0.08	0.275
9	950	0.095	0.37
10	1500	0.15	0.52
11	1800	0.18	0.7
12	1400	0.14	0.84
13	1000	0.1	0.94
14	600	0.06	1

Table E-2. Continued.

Perturbation	Frequency	Probability	CDF
15	0	0	
16	0	0	
17	0	0	
18	0	0	
Total	10000	1	

Table E-3: Case 3 with Low Mean and High Standard Deviation

Perturbation	Frequency	Probability	CDF
1	700	0.07	0.07
2	800	0.08	0.15
3	900	0.09	0.24
4	1000	0.1	0.34
5	1100	0.11	0.45
6	1400	0.14	0.59
7	1700	0.17	0.76
8	1000	0.1	0.86
9	800	0.08	0.94
10	600	0.06	1
11	0	0	
12	0	0	
13	0	0	
14	0	0	
15	0	0	
16	0	0	
17	0	0	
18	0	0	
Total	10000	1	

Table E-4: Case 4 with Low Mean and Low Standard Deviation

Perturbation	Frequency	Probability	CDF
1	0	0	0
2	0	0	0
3	500	0.05	0.05
4	1600	0.16	0.21
5	2400	0.24	0.45
6	3500	0.35	0.8
7	2000	0.2	1
8	0	0	
9	0	0	
10	0	0	
11	0	0	
12	0	0	
13	0	0	

Table E-4 Continued

Perturbation	**Frequency**	**Probability**	**CDF**
14	0	0	
15	0	0	
16	0	0	
17	0	0	
18	0	0	
Total	10000	1	

LIST OF REFERENCES

Achugbue, J. O. and Chin, F. Y. (1979). “The Effectiveness of Output Modification by Rounding for Protection of Statistical Databases.” INFORM 17(3): 209-218.

Adam, N. R. and Jones, D. H. (1989). “Security of Statistical Databases with an Output Perturbation Technique.” Journal of Management Information System 6(1): 101-110.

Adam, N. R. and Wortmann, J. C. (1989). “Security-Control Methods for Statistical Database: A Comparative Study.” ACM Computing Surveys 21(4): 515-556.

Angluin, D. (1988). “Queries and Concept Learning.” Machine Learning 2(4): 319-342.

Angluin, D. and Laird, P. (1988). “Learning from Noisy Examples.” Machine Learning 2(4): 343-370.

Anwar, N. (1993). “Micro-Aggregation – the Small-Aggregates Method.” Internal report. Luxembourg, Eurostat.

Aslam, J.A. and Decatur, S.E. (1993). “General Bounds on Statistical Query Learning and PAC Learning with Noise via Hypothesis Boosting.” In Proceedings of the 34rd Annual IEEE Symposium on Foundations of Computer Science (FOCS ‘93), Palo Alto, California: 282-291.

Beck, L. L. (1980). “A Security Mechanism for Statistical Database.” ACM Transactions on Database Systems 5(3): 316-338.

Blum, A., Furst, M. Jackson, J. Kearns, M.J. Mansour, Y. and Rudich, S. (1994). “Weakly Learning DNF and Characterizing Statistical Query Learning Using Fourier Analysis.” In Proceedings of the 26th Annual ACM Symposium on the Theory of Computing (STOC ‘94), Montréal, Canada: 253-262.

Blumer, A., Ehrenfeucht, A., Haussler, D., and Warmuth, M. K. (1987). “Occam’s Razor.” Information Processing Letters 24(6): 377-380.

Blumer, A., Ehrenfeucht, A., Haussler, D., and Warmuth, M. K. (1989). “Learnability and the Vapnik-Chervonenkis Dimension.” Journal of the ACM 36(4): 929-965.

Brand, R. (2002). “Microdata protection through noise addition. Inference Control in Statistical Databases.” Berlin Heidelberg, Springer. 2316: 97-116.

Brankovic, L., Miller, M., Horak, P. and Wrightson, G. (1997). “Usability of Compromise-free Statistical Databases for Range Sum Queries.” In Proceedings of 9th International Conference on Scientific and Statistical Database Management (SSDBM ‘97), Olympia, Washington: 144-154.

Bshouty, N. H. (1998). “A New Composition Theorem for Learning Algorithms.” In Proceedings of the 30th Annual ACM Symposium on the Theory of Computing (STOC ’98), Dallas, Texas: 583-589.

Bshouty, N. H., Jackson, J., and Tamon, C. (2003). “Uniform-Distribution Attribute Noise Learnability.” Information and Computation 187(2): 277-290.

Cesa-Bianchi, N., Dichterman, E., Fischer, P., Shamir, E., and Simon, H.U. (1999). “Sample-Efficient Strategies for Learning in the Presence of Noise.” Journal of the ACM 46(5): 684-719.

Chen, G. and Keller-McNulty, S. (1998). “Estimation of Identification Disclosure Risk in Microdata.” Journal of Official Statistics 14: 79-95.

Chin, F. Y., Kossowski, P., and Loh, S. C. (1984). “Efficient Inference Control For Range Sum Queries.” Theoretical Computer Science 32:77-86.

Chin, F. Y. and Ozsoyoglu, G. (1979). “Security in Partitioned Dynamic Statistical Databases.” In Proceedings of the IEEE International Computer Software and Applications Conference (COMPSAC ’79), Chicago, Illinois: 594-601.

Chin, F. Y. and Ozsoyoglu, G. (1981). “Statistical Database Design.” ACM Transactions on Database Systems 6(1): 113–139.

Chin, F. Y. and Ozsoyoglu, G. (1982). “Auditing and Inference Control in Statistical Databases.” IEEE Transaction Software Engineering 8(6): 574-582.

Chu, P. C. (1997). “Cell Suppression Methodology: The Importance of Suppressing Marginal Totals.” IEEE Transactions on Knowledge and Data Engineering 9(4): 513-523.

Cox, L.H. (1975). “Disclosure Analysis and Cell Suppression.” In Proceedings of the American Statistical Association (Social Statistics Section): 750-755.

Cox, L.H. (1980). “Suppression Methodology and Statistical Disclosure Control.” Journal of American Statistical Association (Theory and Methods Section) 75(370): 377-385.

Crises G. (2004). “Additive Noise for Microdata Privacy Protection in Statistical Databases.” Research Report.
http://vneumann.etse.urv.es/publications/reports/additivenoise.pdf
(accessed July 2005)

Crises, G. (2004a). “An Introduction to Microdata Protection for Database Privacy.” Research Report. http://vneumann.etse.urv.es/publications/reports/microdata_introduction.pdf (accessed July 2005)

Crises, G. (2004b). “Synthetic Microdata Generation for Database Privacy Protection.” Research Report. http://vneumann.etse.urv.es/publications/reports/synthetic_methods.pdf (accessed July 2005)

Crises, G. (2004c). “Non-Perturbative Methods for Microdata Privacy in Statistical Databases.” Research Report. http://vneumann.etse.urv.es/reports/nonperturbative_methods.pdf (accessed July 2005)

Crises, G. (2004d). “Perturbation Masking for Microdata Privacy Protection in Statistical Databases.” Research Report. http://vneumann.etse.urv.es/reports/perturbative_methods.pdf (accessed July 2005)

Crises, G. (2004e). “Trading Off Information Loss and Disclosure Risk in Database Privacy Protection.” Research Report. http://vneumann.etse.urv.es/publications/reports/combining.pdf (accessed July 2005)

Cristianini, N. and Shawe-Taylor, J. (2000). An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge, Cambridge University Press.

Dalenius, T. (1981). “A Simple Procedure for Controlled Rounding.” Statistik Tidskrift 3: 202-208.

Dalenius, T and Reiss, S.P. (1982). “Data-swapping: A Technique for Disclosure Control.” Journal of Statistical Planning and Inference 6: 73–85.

Decatur, S. E. (1996). “Learning in Hybrid Noise Environments Using Statistical Queries.” In Learning From Data: Artificial Intelligence and Statistics V. Edited by Fisher, V. D. and Lenz, H.J. New York, Springer Verlag: 259-270.

Decatur, S.E. (1997). “PAC Learning with Constant-Partition Classification Noise and Applications to Decision Tree Induction.” In Proceedings of the 6th International Workshop on Artificial Intelligence and Statistics, Fort Lauderdale, Florida: 147-156.

Decatur, S. E. and Gennaro, R. (1995). “On Learning from Noisy and Incomplete Examples.” In Proceedings of the 8th Annual ACM Conference on Computational Learning Theory (COLT ,95), Santa Cruz, California: 353-360.

Defays, D. and Nanopoulos, P. (1993). “Panels of Enterprises and Confidentiality: the Small Aggregates Method.” In Proceedings of 92 Symposium on Design and Analysis of Longitudinal Surveys, Ottawa, Canada: 195-204.

Denning, D. E. (1980). “Secure Statistical Databases with Random Sample Queries.” ACM Transactions on Database Systems 5(3): 291-315.

Denning, D. E. (1983). “A security Model for the Statistical Database Problem.” In Proceedings of the 2nd International Workshop on Statistical Database Management (SSDBM ’83), Los Altos, California: 368-390.

Denning, D. E., Denning, P. J. and Schwartz, M. D. (1979). “The Tracker: A Threat to Statistical Database Security.” ACM Transactions on Database Systems 4(1): 76-79.

Denning, D. E. and Schlorer, J. (1980). “A Fast Procedure for Finding a Tracker in A Statistical Database” ACM Transactions on Database Systems 5(1): 88-102.

Denning, D. E. and Schlorer, J. (1983). “Inference Control for Statistical Databases.” Computer 16(7): 69–82.

Denning, D. E., Schlorer, J., and Wehrle, E. (1982). Memoryless Inference Controls for Statistical Satabases. Purdue University.

DeWaal, A. G. and Willenborg, L. C. R. J. (1995). “Global Recordings and Local Suppressions in Microdata Sets.” In Proceedings of Statistics Canada Symposium 95, Ottawa, Canada: 121–132.

Dinur, I. and Nissim, K. (2003). “Revealing Information while Preserving Privacy.” ACM Press 9(12): 202–210.

Dobkin, D., Jones, A. K., and Lipton, R. J. (1979). “Secure Databases: Protection Against User Influence.” ACM Transactions on Database Systems 4(1): 97-106.

Domingo-Ferrer, J. and Mateo-Sanz, J. M. (2002). “Practical Data-oriented Microaggregation for Statistical Disclosure Control.” IEEE Transactions on Knowledge and Data Engineering 14(1):189–201.

Domingo-Ferrer, J., Mateo-Sanz J. and Torra, V. (2001). “Comparing SDC Methods for Microdata on the Basis of Information Loss and Disclosure Risk.” Pre-proceedings of Exchange of Technology and Know-how. and. New Techniques and Technologies for Statistics (ETK-NTTS ‘01), Crete, Greece. 2: 807-826.

Domingo-Ferrer, J. and Mateo-Sanz J. (1998). “Current Directions in Statistical Data Protection.” Research in Official Statistics 2: 105-112.

Domingo-Ferrer, J. and Torra, V. (2001). “A Quantitative Comparison of Disclosure Control Methods for Microdata.” Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies. Edited by P Doyle, P., Lane, J., Theeuwes, J. and Zayatz, L. Amsterdam, North-Holland: 111-134.

Domingo-Ferrer, J. and Torra, V. (2003). “On the Connections Between Statistical Disclosure Control for Microdata and Some Artificial Intelligence Tools.” Information Sciences 151: 153–170.

Duncan, G. T. and Fienberg S. E. (1999). “Obtaining Information While Preserving Privacy: a Markov Perturbation Method for Tabular Data.”. In Proceedings of Statistical Data Protection, Lisbon. Luxembourg, Eurostat: 351-362.

Duncan, G. T., Fienberg S. E., Krishnan, R., Padman, R. and Roehrig, S. F. (2001). “Disclosure Limitation Methods and Information Loss for Tabular Data.” In Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies. Edited by P Doyle, P., Lane, J., Theeuwes, J. and Zayatz, L. Amsterdam, North-Holland:135-166.

Duncan, G. T., Keller-McNulty, S. A. and Stokes, S. L. (2004). “Database Security and Confidentiality: Examining Disclosure Risk vs. Data Utility through the R-U Confidentiality Map.”
http://www.niss.org/technicalreports/tr142.pdf
(accessed July 2005)

Duncan, G. T. and Lambert, D. (1989). “The Risk of Disclosure of Microdata.” Journal of Business and Economic Statistics 7: 207-17.

Fellegi, I. P. (1972). “On the Question of Statistical Confidentiality.” Journal of American Statistical Association 67(337): 7-18.

Fellegi, I. P. and Phillips, J. L. (1974). “Statistical Confidentiality: Some Theory and Applications to Data Dissemination.” Annals of Economic and Social Measurement 3(2): 399-409.

Felso, F., Theeuwes, J. and Wanger G. G. (2001). “Disclosure Limitation Methods in Use: Results of a Survey.” In Confidentiality, Disclosure and Data Access: Theory and Practical Applications for Statistical Agencies. Edited by P Doyle, P., Lane, J., Theeuwes, J. and Zayatz, L. Amsterdam, North-Holland: 17-42.

Fienberg, S. E. and McIntyre, J. (2004). Data swapping: Variations On A Theme By Dalenius and Reiss. Privacy in Statistical Databases. Berlin Heidelberg, Springer. 3050: 14–29.

Friedman, A. D., and Hoffman, L. J. (1980). “Towards A Fail-safe Approach to Secure Databases.” In Proceedings of IEEE Symposium on Security and Privacy, Oakland, California: 18-22.

Garfinkel, R., Gopal, R., Goes, P. (2002). “Privacy Protection of Binary Confidential Data Against Deterministic, Stochastic, and Insider Threat.” Management Science 48(6): 749-764.

Garfinkel, R., Gopal, R. and Rice, D. (2004). “New Approaches to Disclosure Limitation While Answering Queries to a Database: Protecting Numerical Confidential Data Against Insider Threat Based on Data or Algorithms.”
http://www-eio.upc.es/seminar/04/garfinkel.pdf
(accessed July 2005)

Goldman, S. A. (1991). Computational Learning Theory. Lecture Notes.
http://www.cs.wustl.edu/cs/cs/archive/CS582_SP96/
(accessed July 2005)

Goldman, S. A. and Sloan, R. (1995). “Can PAC learning Algorithms Tolerate Random Attribute Noise?” Algorithmica 14(1): 70-84.

Gomatam, S., Karr, A. F., Reiter, J. P. and Senil, A. P. (2004). “Data Dissemination and Disclosure Limitation in a World Without Microdata: A Risk-Utility Framework for Remote Access Analysis Servers.”
http://www.niss.org/technicalreports/tr138.pdf
(accessed July 2005)

Gopal, R., Garfinkel, R. and Goes, P. (2000). “Confidentiality via Camouflage: The CVC Approach to Disclosure Limitation When Answering Queries to Databases.” Operations Research 50(3): 501-516.

Gopal, R., Goes, P. and Garfinkel, R. (1998). “Interval Protection of Confidential Information in a Database.” Journal on Computing 10(3): 309-322.

Hansen, S. L. and Mukherjee. S. (2003). “A Polynomial Plgorithm for Pptimal Univariate Microaggregation.” IEEE Transactions on Knowledge and Data Engineering 15(4): 1043-1044.

Haq, M. I. (1977). “On Safeguarding Statistical Disclosure by Giving Approximate Answers to Queries.” In Proceedings of International Computer Symposium, Liège, Belgium: 491-495.

Haq, M. I. (1975). “Insuring Individual’s Privacy from Statistical Database Users.” In Proceedings of National Computer Conference, Anaheim, California. 44: 941-946.

Hoffman, L. J. (1977). Modern Methods for Computer Security and Priuacy. New Jersey, Prentice-Hall, Englewood Cliffs.

Hoffman, L. J., and Miller, W. F. (1970). “Getting A Personal Dossier From a Statistical Data Bank.” Datamation 16(5): 74-75.

Holvast, J. (1999). "Statistical Dissemination, Confidentiality and Disclosure." In Proceedings of the Joint Eurostat/UNECE Work Session on Statistical Data Confidentiality. Luxembourg, Eurostat: 191-207.

ILOG (1999). ILOG CPLEX 6.5 User's Manual.

Jackson, J. (2003). "On the Efficiency of Noise-Tolerant PAC Algorithms Derived from Statistical Queries." Annals of Mathematics and Artificial Intelligence 39(3): 291-313.

Jaro, M. A. (1989). "Advances in Record-linkage Methodology as Applied to Matching the 1985 Census of Tampa, Florida." Journal of American Statistical Association 84: 414-420.

Jonge, W. DE. (1983). "Compromising statistical databases: Responding to Queries about Means." ACM Transactions on Database Systems 8(1): 60-80.

Kearns, M. (1993). "Efficient Noise-Tolerant Learning from Statistical Queries." In Proceedings of the 25th Annual ACM Symposium on Theory of Computing, San Diego, California: 392-401.

Kearns, M. and Li, Ming. (1993). "Learning in the Presence of Malicious Errors." Journal on Computing 22(4): 392-401.

Kelly, J.P., Golden, B.L., and Assad, A.A. (1992). "Cell Suppression: Disclosure Protection for Sensitive Tabular Data." Networks 22: 397-417.

Kim, J. J. (1986). "A Method for Limiting Disclosure in Microdata Based on Random Noise and Transformation." In Proceedings of the 2nd on Survey Research Methods, Alexandria, Virginia: 303-308.

Kleinberg, J., Papadimitriou, C. and Raghavan, P. (2000). "Auditing boolean attributes." In Proceedings of the 9th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, Dallas, Texas: 86-91.

Kooiman, P., Willenborg, L., and Gouweleeuw, J. (1998). "A Method of Disclosure Limitation of Microdata." Research Report. Statistics Netherlands.

Lambert, D. (1993). "Measures of Disclosure Risk and Harm." Journal of Official Statistics 9: 461–8.

Lefons, D., Silverstri, A., and Tangorra, F. (1983). "An Analytic Approach to Statistical Databases." In Proceedings of 9th Conference on Very Large Databases, Florence, Italy: 260-273.

Leiss, E. (1982). "Randomizing a Practical Method for Protecting Statistical Databases against Compromise." In Proceedings of 8th Conference on Very Large Databases, Mexico City. Mexico: 189-196.

Li, Y., Wang, L., Wang, X.S. and Jajodia, S. (2002a). “Auditing Interval-based Inference.” In Proceedings of the14th Conference on Advanced Information Systems Engineering (CAiSE ’02), Toronto, Canada: 553-568.

Li, Y., Wang, L., Zhu, S.C. and Jajodia, S. (2002b). “A Privacy Enhanced Microaggregation Method.” In Proceedings of the 2nd International Symposium on Foundations of Information and Knowledge Systems (FoIKS ‘02), Salzau Castle, Germany: 148–159.

Liew, C. K., Choi, W. J., and Liew, C. J. (1985). “A Data Distortion by Probability Distribution.” ACM Transactions on Database Systems 10(3): 395-411.

Luige, T. and Meliskova, J. (1999). “Confidentiality Practices in the Trnsition Countries.” In Proceedings of the Joint Eurostat/UNECE Work Session on Statistical Data Confidentiality, Luxembourg, Eurostat: 287-319.

Malvestuto, F.M. (1993). “A Universal Écheme Approach to Statistical Databases Containing Homogeneous Summary Tables.” ACM Trans. Database Systems 18(4): 678-708.

Malvestuto, F.M. and Mezzini, M. (2003). “Auditing Sum Queries.” In Proceedings of the 9th International Conference on Database Theory (ICDT ’03), Siena, Italy: 126–146.

Malvestuto, F.M. and Moscarini, M. (1990). “Query Evaluability in Statistical Databases.” IEEE Transactions on Knowledge and Data Engineering 2(4): 425-430.

Malvestuto, F.M. and Moscarini, M. (1998). “Computational Issues Connected with the Protection of Sensitive Statistics by Auditing Sum-queries.” In Proceedings of IEEE Scientific and Statistical Database Management, Capri, Italy: 134–144.

Malvestuto, F.M., Moscarini, M. and Rafanelli, M. (1991). “Suppressing Marginal Cells to Protect Sensitive Information in a Two-Dimensional Statistical Table.” In Proceedings of the10th ACM SIGACT-SIGMOD-SIGART Symposium Principles of Database Systems, Denver, Colorado: 252-258.

Más, M. (2000). “Statistical Data Protection Techniques.”
http://www.eustat.es/document/datos/prot_seguridat_i.pdf
(assessed July 2005)

Mateo-Sanz, J. M. and Domingo-Ferrer. J. (1999). “A Method for Data-Oriented Multivariate Microaggregation.” In Proceedings of Statistical Data Protection '98, Luxembourg, UK: 89-99.

Matloff, N. E. (1986). “Another Look at the Use of Noise Addition for Database Security.” In Proceedings of IEEE Symposium on Security and Privacy, Oakland, California: 173-180.

Muralidhar, K., Batra, D. and Kirs, P. J. (1995). “Accessibility, Security, and Accuracy in Statistical Database: The Case for the Multiplicative Fixed Data Perturbation Approach.” Management Science 41(9): 1549-1564.

Muralidhar, K., Parsa, R., and Sarathy, R. (1999). “A General Additive Data Perturbation

Method for Database Security.” Management Science 45(10): 1399-1415.

Muralidhar, K., Li, H. and Sarathy, R. (2004). “Disclosure Risk Problems with Confidentiality via Camouflage.”

Natarajan, B. K. (1991). Machine Learning: A Theoretical Approach. San Francisco, California, Morgan Kaufmann Publishers, Inc.

Oganian, A. (2002). Security and Information Loss in the Protection of Statistical Databases. Dissertation Thesis, Universitat Politécnica de Catalunya.

Ozsoyoglu, G. and Chin, F. Y. (1982). “Enhancing the Security of Statistical Database with a Question-Answering System and a Kernel Design.” IEEE Transactions in Software Engineering 8(3): 223-234.

Pagliuca, D. and Seri, G. (1998). “Some Results of Individual Ranking Method on the System of Enterprise Accounts Annual Survey.” Esprit SDC Project, Deliverable MI-3/D2.

Reiss, S.P. (1984). “Practical Data-swapping: The First Steps.” IEEE Transactions on Database Systems 9: 20-37.

Samuel, S. M. (1998). “A Bayesian, Speices-Sampling-Inspired Approach to the Unique Problem in Microdata Disclosure Risk Assessment.” Journal of Official Statistics 14: 373-383.

Sande, G. (1983). “Automated Cell Suppression to Reserve Confidentiality of Business Statistics.” In Proceedings of the 2nd International Workshop on Statistical Database Management, Los Altos, California: 346-353.

Sarathy, R. and Muralidhar, K. (2002). “The Security of Confidential Numerical Data in Databases.” Information Systems Research 13(4): 389-403.

Schlorer, J. (1975). “Identification and Retrieval of Personal Records From a Statistical Data Bank.” Methods of Information in Medicine: 14(1): 7-13.

Schlorer, J. (1976). “Confidentiality of Statistical Records: A Threat Monitoring Scheme of On-line Dialogue.” Methods of Information in Medicine 15(1): 36-42.

Schlorer, J. (1980). “Disclosure From Statistical Databases: Quantitative Aspects of Trackers.” ACM Transactions on Database Systems 5(4): 467-492.

Schlorer, J. (1981). “Security of Statistical Databases: Multidimensional Transformation.” ACM Transactions on Database Systems 6(1): 95-112.

Schlorer, J. (1983). “Information Loss in Partitioned Statistical Databases.” Computer Journal 26(3): 218-223.

Schwartz, M. D., Denning, D. E., and Denning, P. J. (1979). “Linear Queries in statistical Databases.” ACM Transactions on Database Systems 4(2): 156-167.

Schölkopf, B. and A. J. Smola. (2001). Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond. Boston, MIT Press.

Sebé, F., Domingo-Ferrer, J., Mateo-Sanz and Torra, V. (2002). “Postmasking Optimization of the Tradeoff between Information Loss and Disclosure Risk in Masked Microdata Sets.” Inference Control in Statistical Databases. Berlin Hedelberg, Springer. 2316: 63-171.

Shackelford, G. and Volper, D. (1988). “Learning k-DNF with Noise in the Attributes.” In Proceedings of the 1988 Workshop on Computing Learning Theory, MIT, Massachusetts: 97-103.

Skinner, C., Marsh, C., Openshaw, S., and Wymer, C. (1994). “Disclosure Control for Census Microdata.” Journal of Official Statistics 10: 31-51.

Sloan, R. (1988). “Types Noise in Data for Concept Learning.” In the 1988 Workshop on Computational Learning Theory, MIT, Massachusetts: 91-96.

Sloan, R. (1995). “Four Types of Noise in Data for PAC Learning.” Information Processing Letter 54: 157-162.

Spruill, N. L. (1983). “The Confidentiality and Analytic Usefulness of Masked Business Microdata.” In Proceedings of the Section on Survey Research Methods, American Statistical Association: 602-607.

Sullivan, G.R. (1989). The Use of Added Error to Avoid Disclosure in Microdata Release. Dissertation Thesis, Iowa State University.

Sullivan, C. M. (1992). “An Overview of Disclosure Principles.” Bureau of the Census Statistical Research Division Research Report Series No. RR-92/09.

Tendick, P. (1991). “Optimal Noise Addition for Preserving Confidentiality in Multivariate Data.” Journal of Statistical Planning and Inference 27(2): 341-353.

Tendick, P. and Matloff, N. (1994). “A modified Random Perturbation Method for Database Security.” ACM Transactions on Database Systems 19(1): 47-63.

Trottini, M. and Fienberg, S. (2002). “Modeling User Uncertainty for Disclosure Risk and Data Utility.” International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 10(5): 511-528.

Truta, T. M., Fotouhi, F. and Barth-Jones, D. (2004). “Disclosure Risk Measures for the Sampling Disclosure Control Method.” In Proceedings of ACM 2004 Symposium on Applied Computing, Nicosia, Cyprus: 301-306.

Valiant, L. G. (1984). “A Theory of the Learnable.” Communications of the ACM 27: 1134-1142.

Valiant, L. G. (1985). “Learning Disjunctions of Conjunction.” In Proceedings of 9th International Joint Conference on Artificial Intelligence, Los Angeles, California:

Vapnik, V. (1998). Statistical Learning Theory. New York, John Wiley & Sons.

Vapnik, V. and Chervonenkis, A. (1971). “On the Uniform Convergence of Relative Frequencies of Events to their Probabilities.” Theory of Probability and its Applications 16(2): 264-280.

Willenborg, L. and Waal, T. (2000). Elements of Statistical Disclosure Control. New York. Springer.

Willenborg, L. and Waal, T. (1996). Statistical Disclosure Control in Practice. New York, Springer.

Yancey, W. E., Winkler, W. E. and Creecy, R. H. (2002). “Disclosure Risk Assessment in Perturbative Microdata Protection.” Inference Control in Statistical Databases, Berlin Hedelberg, Springer. 2316: 135-152.

Yu, C. T., and Chin, F. Y. (1977). “A Study on the Protection of Statistical Databases.” In Proceedings of ACM SIGMOD International Conference on Management of Data, Toronto, Canada: 169-181.

BIOGRAPHICAL SKETCH

Ling He graduated from the University of International Business and Economics with a Bachelor of Arts degree in economics in 1996. She received a Master of Science degree in Decision and Information Sciences in 2003 and a Ph.D. degree in 2005 at the University of Florida. Her research interests focus on database management, machine learning theory and applications, statistical learning theory, data-mining, information security, information retrieval, and e-Commerce.

She intends to pursue an academic research and teaching career after the completion of her doctoral degree.