December 2004

A Data Perturbation Approach to Privacy Protection in Data Mining

Xiao-Bai Li
University of Massachusetts, Lowell

Sumit Sarkar
University of Texas at Dallas

A DATA PERTURBATION APPROACH TO PRIVACY PROTECTION IN DATA MINING

Xiao-Bai Li
College of Management
University of Massachusetts, Lowell
Lowell, MA U.S.A.
bob_li@uml.edu

Sumit Sarkar
School of Management
University of Texas at Dallas
Richardson, TX U.S.A.
sumit@utdallas.edu

Abstract

Advances in data mining techniques have raised growing concerns about privacy of personal information. Organizations that use their customers' records in data mining activities are forced to take actions to protect the privacy of the individuals involved. A common practice for many organizations today is to remove the identity-reated attributes from customer records before releasing them to data miners or analysts. In this study, we investigate the effect of this practice and demonstrate that a majority of the records in a dataset can be uniquely identified even after identity related attributes are removed. We propose a data perturbation method that can be used by organizations to prevent such unique identification of individual records, while providing the data to analysts for data mining. The proposed method attempts to preserve the statistical properties of the data based on privacy protection parameters specified by the organization. We show that the problem can be solved in two phases, with a linear programming formulation in phase one (to preserve the marginal distribution), followed by a simple Bayes-based swapping procedure in phase two (to preserve the joint distribution). The proposed method is compared with a random perturbation method in classification performance on two real-world datasets. The results of the experiments indicate that it significantly outperforms the random method.

Keywords: Privacy, data mining, linear programming, Bayesian method

Introduction

In recent years, we have observed an explosion of digital data generated and collected by individuals and organizations. The widespread use of computers and the Internet for transaction processing, and the advances in storage technology and database systems, have allowed us to generate and store mountains of data. In tandem with this unprecedented growth of technologies to collect and store data, techniques for data mining have emerged and flourished in order to extract useful knowledge from massive volumes of data. Because of its ability in discovering hidden patterns in the transaction and customer data, data mining has been applied to a wide variety of domains, including database marketing, credit and loan evaluation, fraud detection, customer profiling, Web usage analysis, and medical diagnostics.

While successful applications of data mining are encouraging, there are increasing concerns about invasions and potential threats to privacy of personal information by information technology in general, and by data mining in particular. A survey by Time/CNN (Greengard 1996) revealed that 93 percent of respondents believed companies selling personal data should be required to gain permission from the individuals. In another study (Culnan 1993), more than 70 percent of participants responded negatively to questions related to the secondary use of private information. In 1990, Lotus attempted to release a CD-ROM with data on about 100 million households in the United States. However, the product generated such strong public protests regarding privacy issues that Lotus was forced to withdraw the project (Rotenberg 1992). Recent studies point to similar and growing concerns with privacy (Stanford 2002; Wang et al. 1998).

In order to resolve the conflict between data mining and privacy protection, researchers in the data mining community have proposed various methods. Agrawal and Srikant (2000) considered building a decision tree classifier from data where the confidential values have been perturbed. By using a distribution reconstruction procedure, the authors were able to build classifiers whose accuracy is comparable to that of classifiers built with the original data. Estivill-Castro and Brankovic (1999) proposed a data swapping method, also used in decision trees for classification. This method aims at maintaining the same decision tree structure while perturbing the data. It does not consider the statistical distribution in the data. Therefore, the perturbed data may produce poor results when used for other data mining tasks, or even for different decision tree classifiers. Evfimievski et al (2002) presented a framework for mining association rules from transaction data that have been randomized to preserve individuals' privacy. The authors derived a support estimator, which allows a data miner to recover itemset supports from randomized data and subsequently discover association rules from the data. Other studies on privacy preserving mining of association rules can be found in Atallah et al. (1999) and Verykios et al. (2004). Many of the studies described above approach the privacy issue from a data miner's standpoint. They focus on how to develop algorithms for mining those datasets where confidential values are deleted or perturbed due to privacy concerns. We believe, however, it is more important to approach the issue from the standpoint of an organization that owns data, because the primary concern of a data miner is to discover useful knowledge from the data, while an organization has to set privacy protection as its first priority. In this study, therefore, we focus on the problem of how to preserve privacy while still enabling meaningful data mining that benefits the organization.

The issue of protecting confidential data is not new. There has been extensive research in the area of statistical databases (SDBs) on how to provide summary statistical information without disclosing individuals' confidential data. The privacy issue arises in SDB when summary statistics are derived on very few (or a single) individuals' data. In this case, releasing the summary statistics leads to disclosure of individual confidential data. The methods for preventing such disclosure can be broadly classified into two categories: query restriction, which prohibits queries that would reveal confidential data; and data perturbation, which alters individual data in a way such that the summary statistics remain approximately the same. Adam and Workman (1989) presented an excellent survey of these methods. Duncan and Mukherjee (2000) investigated the effectiveness of the query restriction and data perturbation methods against tracker attacks to online SDBs. Using a mathematical programming formulation, the study showed that a combination of query restriction and data perturbation (termed *data masking* in the article) provides better protection than when these methods are used separately. In general, both query restriction and data perturbation methods have been extensively investigated and employed (see, for example, Cox 1995; Muralidhar et al. 1999).

The privacy issues in data mining are somewhat different from those in SDBs. The main purpose of an SDB is to provide summary statistics, while data mining involves tasks such as classification and mining association rules, in addition to providing summary statistics. In an SDB, a user normally cannot retrieve a complete relational table and can use only a few limited query types, often restricted to access aggregate statistics, to retrieve information. In data mining, relational (not contingency) tables containing individual records have to be released to data miners in order to perform the above data mining tasks. Therefore, query restriction methods are no longer applicable and data perturbation becomes the primary approach for privacy protection in data mining. In addition, many data mining tasks involve processing and analyzing categorical attributes (e.g., the class attribute in classification and all attributes in association rules mining). Although there are studies that deal with categorical data in the SDB research (Chowdhury et al. 1999; Cox 1995; Fienberg et al. 2000), most of them are directed at data presented in a summarized contingency table. Garfinkel et al. (2002) proposed a privacy protection method that applies to data in a relational table, but the method is limited to cases where confidential attributes are binary.

In this study, we investigate the privacy problem where individual records in a dataset can be uniquely identified without using identity-related attributes. We propose a data perturbation method that can be used by organizations to prevent deterministic disclosure of individuals' confidential information, while providing the data to data miners or analysts for data mining. The proposed method attempts to preserve the statistical properties of the data based on privacy protection parameters specified by the organization. The basic idea is to maintain statistical distributions via appropriate data swapping (we use the terms *perturbation* and *swapping* interchangeably in this paper, although their connotations are somewhat different in the related literature). We show that the problem can be solved in two phases: in phase one a linear programming formulation is used to preserve the marginal distribution, and in phase two a simple Bayes-based swapping procedure is employed to preserve the joint distribution. The proposed method applies to situations where confidential attributes are categorical (either as originally represented, or converted from numeric values).

The Privacy Protection Problem

Typically, there are three parties involved in the privacy problem in data mining: (1) the *data owner* (which is the organization that owns the data) who has complete privileges to access the data and wants to discover knowledge from the data to gain competitive advantage, without compromising the confidentiality of the data; (2) *individuals* who provide their personal information to the data owner and want their privacy protected; and (3) the *data miner* (insider or outsider) who performs data mining for the data owner with the given data and who is regarded as a potential data snooper in this study.

There is a common misconception, in many organizations, that if the identity-related attributes, such as social security number, name, and phone number, are removed from the released data, there will be no leak in individuals' confidential data to the third party. The fact is, a dataset with identity related attributes removed would still contain many unique records, which can be often identified by a method such as a GROUP BY query or a sorting algorithm. With some additional information, the data snooper can easily find an individual's confidential data. To demonstrate the problem, let us look at a hypothetical, but realistic, example.

A life insurance company wants to know the relationship between the amount of death benefit (confidential) and a set of demographic attributes (non-confidential), in order to launch an effective marketing initiative. A consultant is hired to conduct such analysis. The company provides him with a dataset consisting of its 100,000 customer records. To prevent disclosure of confidential data, attributes such as account number, name, and phone number are deleted. In addition, the values of some attributes are grouped. For example, the amount of death benefit is grouped into a few interval categories; the location attribute only shows which state the customer resides in, instead of the complete street address. The data released to the consultant includes the death benefit amount (grouped) and six demographic attributes: age (5), gender (2), location (50), education (5), occupation (10), and marital status (5), where the number of categories in each attribute is shown in parentheses. The company would believe that no confidential data could be disclosed from this processed dataset. A simple calculation, however, shows a different picture. The total number of category combinations for the six attributes is $5 \times 2 \times 50 \times 5 \times 10 \times 5 = 125{,}000$. Assuming each category combination is presented with equal likelihood, then each of the 100,000 customers will be a unique record. In real situations, of course, the assumption of the uniform distribution across all dimensions is rather unrealistic. More likely, some customers will share the same attribute values, but others will have unique values. These unique individuals will be exposed to the risk of confidentiality disclosure. Suppose the consultant has noted that there is a unique record with {age = 60–69, gender = female, location = FL, education = bachelor, occupation = retired, marital status = widowed}, which matches the demographic data of one of his relatives. If he knew this relative has an account with the company, then he has effectively discovered the amount of his relative's death benefit. This disclosure could have serious financial, legal, or even criminal implications.

In order to formulate this privacy protection problem more rigorously, we first define some terms. We assume that there is only one confidential attribute in the data, but the definitions below can be easily extended to cases with multiple confidential attributes. The definitions are explained using a reduced life insurance example shown in Table 1, which has one confidential attribute (Amount) and three non-confidential attributes.

> **Definition 1**: *A **full pattern** is a category combination that involves all attributes. A **non-confidential pattern** is a category combination that involves all non-confidential attributes.*

For example, the category combination {Age = 30–39, Gender = Female, Location = CA, Amount = Med}, as shown in record #1, is a full pattern, while the part {Age = 30–39, Gender = Female, Location = CA} is a non-confidential pattern.

> **Definition 2**: *A record is **identifiable** if its full pattern can be completely determined by its non-confidential pattern. A record is **uniquely identifiable** if it is identifiable and there does not exist another record that has the same full pattern. A group of records are **collectively identifiable** if each member record is identifiable and all members of the group have the same full pattern; in this case, each member of the group is also said to be collectively identifiable.*

In Table 1, a record marked with a U is uniquely identifiable, and a record with V is collectively identifiable (V1, V2 and V3 denote three different groups). The unmarked records are unidentifiable. We use the following notation to facilitate the formulation of the problem:

$\mathbf{X}$ $(N \times J)$	the complete set of non-confidential data matrix, with N records and J attributes, which is not subject to perturbation;
$\mathbf{y}$, $\tilde{\mathbf{y}}$ $(N \times 1)$	the original and perturbed confidential attribute, respectively;

Table 1. An Illustrative Example

No.	Age	Gender	Location	Amount	Identifiable Status
1	30–39	Female	CA	Med	U
2	30–39	Female	NY	Low	V1
3	30–39	Female	NY	Low	V1
4	30–39	Male	CA	Med	V2
5	30–39	Male	CA	Med	V2
6	30–39	Male	NY	High	U
7	40–49	Female	CA	Med	U
8	40–49	Female	NY	Med	
9	40–49	Female	NY	High	
10	40–49	Male	CA	Low	U
11	40–49	Male	NY	High	V3
12	40–49	Male	NY	High	V3
13	50–59	Female	NY	Low	U
14	50–59	Male	CA	Med	
15	50–59	Male	CA	High	
16	50–59	Male	NY	High	U

U the set of all uniquely identifiable records;
V the set of all collectively identifiable records, which has G different groups; and
C the number of categories in the confidential attribute.

In this study, we assume the data owner's policy is to prevent deterministic disclosure of confidentiality by perturbing a portion of identifiable records. For records in **U**, this can be achieved by using a specified proportion for perturbation. For records in **V**, if each identifiable group has at least one member record perturbed, then the group is no longer identifiable. For ease of discussion, we assume that exactly one member record in each group is perturbed. The confidential value of an unidentifiable record will not be perturbed since it cannot be determined by its non-confidential values. The objective is to maintain the joint distribution of all attributes, while satisfying the privacy protection policy set by the data owner. It is usually not possible to completely preserve the joint distribution. Since the marginal distribution of the confidential attribute is used in summary statistics, preserving this becomes an important goal in itself. The problem, then, is

Minimize
(G1) The distance between the original and perturbed marginal distributions in the confidential attribute **y**; and

(G2) The distance between the original and perturbed joint distributions;

Subject to
(C1) **U** is perturbed with proportion p, a parameter determined by the data owner;
(C2) For each collectively identifiable group in **V**, exactly one member record (or a specified number or proportion of records) is perturbed.

In addition, none of the unidentifiable records may be changed. Note that solutions to satisfy (C1) and (C2) can always be found. The reason for including the proportion p in (C1)—and similarly the number of member records subject to perturbation in (C2)—as a parameter is that a larger p value will lead to a better protection of confidential data but cause the perturbed distributions to be further away from the original distributions (and thus lead to a less reliable data mining outcome), while a smaller p will have an opposite effect. Therefore, the proportion p is a parameter indicating the trade-off between privacy protection and data mining

(see our discussion for the binary confidential attribute case in the next section, however). Noticing that the above problem involves optimizing multiple objectives, we adopt a two-phase strategy to solve the problem, stated as:

Phase I: Minimize (G1), subject to constraints (C1), and (C2)
Phase II: Minimize (G2), subject to constraints (C1), (C2) and
(C3) The marginal distribution of the perturbed confidential data is preserved during Phase II.

The Approach

Phase I: Preserving the Marginal Distributions

Consider set **U** first. Let N_k (a known quantity) be the number of records in **U** originally having the k^{th} confidential category. Let n_{kh} (a variable) be the number of records in **U** changed from the k^{th} to the h^{th} category; that is, n_{kh} 's are decision variables whose values are to be determined by the Phase I optimization procedure. If the marginal distribution remains the same after perturbation, then

$$\sum_{h=1,...,C;\, h\neq k} n_{kh} = \sum_{h=1,...,C;\, h\neq k} n_{hk},\ k = 1,...,C.$$

If this condition cannot be satisfied, then there exists either a slack (if $\sum n_{kh} < \sum n_{hk}$) or a surplus (if $\sum n_{kh} > \sum n_{hk}$) quantity. Let s_k^- and s_k^+, both nonnegative, be such a slack and a surplus variable, respectively. Then, the Phase I problem can be formulated as a linear programming (LP) problem as follows:

$$\min \quad \sum_{k=1}^{C} (s_k^- + s_k^+), \tag{1a}$$

$$\text{s.t.} \quad \sum_{k=1}^{C} \sum_{h=1,...,C;\, h\neq k} n_{kh} = p \sum_{k=1}^{C} N_k, \tag{1b}$$

$$\sum_{h=1,...,C;\, h\neq k} n_{kh} - \sum_{h=1,...,C;\, h\neq k} n_{hk} + s_k^- - s_k^+ = 0,\ k = 1,...,C, \tag{1c}$$

$$\sum_{h=1,...,C;\, h\neq k} n_{kh} \le N_k,\ k = 1,...,C. \tag{1d}$$

This problem always has a feasible solution since constraint (1b) merely requires the proportion of perturbed records to be *p*, and appropriate values can always be found for slack and surplus variables, s_k^- and s_k^+, to satisfy (1c). When the optimal value of objective function (1a) is zero, the marginal distribution of **y** is completely preserved; otherwise, it is impossible to preserve the marginal distribution. The optimal solution for some n_{kh} may be fractional. In this case, we simply round it to an integer, which should not be an issue since n_{kh} is large in data mining problems.

Now consider set **V**. Based on constraint (C2), there will be *G* records perturbed (one for each group). The set of these *G* records is of the same nature as set **U** with $p = 1$. The method described above can be readily applied to perturbing these *G* records in **V** by setting $p = 1$. When **y** is binary ($C = 2$), the only solution is to change the original value to the other one, and hence no optimization issue is involved. The same is true in perturbing **U** when **y** is binary and *p* is set to one. Doing so is, however, very risky for the data owner because all of the original values can be found by reversing the perturbed values. Therefore, *p* should be set to some value less than 1 when **y** is binary.

Applying formulation (1) to the records in set **U** in the life insurance example, we have the following LP problem (where p is set to 0.5):

$$\begin{aligned}
\min \quad & s_1^- + s_1^+ + s_2^- + s_2^+ + s_3^- + s_3^+ , \\
\text{s.t.} \quad & n_{12} + n_{13} + n_{21} + n_{23} + n_{31} + n_{32} = 3 \ (= 0.5 \times 6) , \\
& n_{12} + n_{13} - n_{21} - n_{31} + s_1^- + s_1^+ = 0 , \\
& n_{21} + n_{23} - n_{12} - n_{32} + s_2^- + s_2^+ = 0 , \\
& n_{31} + n_{32} - n_{13} - n_{23} + s_3^- + s_3^+ = 0 , \\
& n_{12} + n_{13} \le 2 , \\
& n_{21} + n_{23} \le 2 , \\
& n_{31} + n_{32} \le 2 ,
\end{aligned}$$

where subscripts 1, 2, and 3 index the three confidential values and all variables are nonnegative. An optimal solution to this LP problem is $n_{12} = 1$, $n_{23} = 1$, $n_{31} = 1$, and all of the remaining variables, including slack and surplus variables, are 0. Based on this solution, we should perturb one Low record to Med, one Med record to High, and one High record to Low. The optimal objective function value reaches 0, indicating that the marginal distribution of the confidential attribute is preserved. Similarly, an LP problem can be formulated for perturbing records in **V**.

Phase II: Preserving the Joint Distribution

Let $P(\mathbf{X}, \mathbf{y})$ and $\tilde{P}(\mathbf{X}, \mathbf{y})$ be the original and perturbed joint distributions, respectively (here **X** and **y** are random variables, not observed data). Let $D(P(\mathbf{X}, \mathbf{y}), \tilde{P}(\mathbf{X}, \mathbf{y}))$ be the distance between $P(\mathbf{X}, \mathbf{y})$ and $\tilde{P}(\mathbf{X}, \mathbf{y})$, to be minimized in Phase II. Using the well-known I-Divergence (also known as Kullback-Leibler distance) measure, this distance can be defined as

$$D(P(\mathbf{X}, \mathbf{y}), \tilde{P}(\mathbf{X}, \mathbf{y})) = \sum_i P(\mathbf{X}_i, Y_i) \log \frac{P(\mathbf{X}_i, Y_i)}{\tilde{P}(\mathbf{X}_i, Y_i)} \tag{2}$$

where subscript i runs over all possible full patterns. Expression (2) is minimized if and only if (Kullback 1959)

$$P(\mathbf{X}_i, Y_i) = \tilde{P}(\mathbf{X}_i, Y_i), \ \forall i , \tag{3}$$

which is equivalent to

$$P(Y_i \mid \mathbf{X}_i) = \tilde{P}(Y_i \mid \mathbf{X}_i), \ \forall i , \tag{4}$$

since $\tilde{P}(\mathbf{X}_i) = P(\mathbf{X}_i)$ for each *i* (non-confidential data are not perturbed). This condition is unlikely to hold because, for pattern *i*, perturbing its true confidential value y_{it} to a different value y_{ir} will cause $P(Y_i = y_{it} \mid \mathbf{X}_i)$ to decrease and $P(Y_i = y_{ir} \mid \mathbf{X}_i)$ to increase. That is,

$$P(Y_i = y_{it} \mid \mathbf{X}_i) > \tilde{P}(Y_i = y_{it} \mid \mathbf{X}_i), \text{ and} \tag{5a}$$

$$P(Y_i = y_{ir} \mid \mathbf{X}_i) < \tilde{P}(Y_i = y_{ir} \mid \mathbf{X}_i). \tag{5b}$$

Let *I* be the index set for all records available for perturbation. Define two positive quantities for each $i \in I$ as below:

$$\Delta P_i(t) = P(Y_i = y_{it} \mid \mathbf{X}_i) - \tilde{P}(Y_i = y_{it} \mid \mathbf{X}_i), \tag{6a}$$

$$\Delta P_i(r) = \tilde{P}(Y_i = y_{ir} \mid \mathbf{X}_i) - P(Y_i = y_{ir} \mid \mathbf{X}_i). \tag{6b}$$

Our goal now is to select and perturb a set of records, indexed by I_*, whose size is determined by constraints (C1) and (C2), such that

$$\sum_{i \in I_*} \{\Delta P_i(t) + \Delta P_i(r)\} \tag{7}$$

is the minimum among all possible sets that satisfy (C1) and (C2). By definitions (6a) and (6b), we have

$$0 < \Delta P_i(t) < P(Y_i = y_{it} \mid \mathbf{X}_i) \text{ and } 0 < \Delta P_i(r) < 1 - P(Y_i = y_{ir} \mid \mathbf{X}_i).$$

Combining them yields

$$0 < \Delta P_i(t) + \Delta P_i(r) < 1 + P(Y_i = y_{it} \mid \mathbf{X}_i) - P(Y_i = y_{ir} \mid \mathbf{X}_i). \tag{8}$$

This implies that, to minimize expression (7), we should select pattern *i* for which

$$P(Y_i = y_{it} \mid \mathbf{X}_i) - P(Y_i = y_{ir} \mid \mathbf{X}_i) \tag{9}$$

is as small as possible. The intuition behind this criterion is that we should perturb a record's confidential value from its true value to a new one when the conditional probability with the true value is low and that with the new value is high.

To estimate the conditional probability in (9), one would naturally consider the full-order conditional estimator (that makes no assumptions of any kind), which estimates $P(Y_i \mid \mathbf{X}_i)$ based on the i^{th} full and non-confidential patterns in the data. For example, based on records #8 and #9 in Table 1, the full-order estimator yields estimates *P*(High | 40–49, Female, NY) = ½, *P*(Med | 40–49, Female, NY) = ½, and *P*(Low | 40–49, Female, NY) = 0. The estimator is not useful, however, in terms of minimizing the distance between the joint distributions. More precisely, for identifiable patterns, the value associated with criterion (9) using the full-order conditional estimator is constant no matter which, and how, identifiable records are perturbed. In fact, estimating the probabilities for values of confidential attributes of identifiable patterns is problematic since only a single record exists with the corresponding non-confidential pattern. Motivated by this reasoning, we consider using the simple Bayes estimator, which avoids high order computation by assuming conditional independence, to estimate $P(Y_i \mid \mathbf{X}_i)$.

The simple Bayes estimator for $P(Y_i \mid \mathbf{X}_i)$, the *posterior* probability of Y_i given $\mathbf{X}_i$, is given by

$$P(Y_i \mid \mathbf{X}_i) = \frac{P(Y_i)}{P(\mathbf{X}_i)} \prod_{j=1}^{J} P(X_{ij} \mid Y_i), \ \forall i, \tag{10}$$

where X_{ij} is the *j*th attribute of $\mathbf{X}_i$. Substituting expression (10) into (9), we obtain the difference in posterior, due to perturbing record *i*'s confidential value from y_{it} to y_{ir}, as:

$$d_i(t,r) = K_i\{p(y_{it})\prod_{j=1}^{J} p(x_{ij} \mid y_{it}) - p(y_{ir})\prod_{j=1}^{J} p(x_{ij} \mid y_{ir})\}, \tag{11}$$

where *x* and *y* are the observed values of random variables *X* and *Y*; $p(\cdot)$ is the estimate of $P(\cdot)$; and $K_i = 1/\,p(\mathbf{x}_i)$ is a constant (normalizing factor) for the *i*th pattern. The Phase II problem can now be stated as

$$\min \quad \sum_i d_i(t,r), \tag{12a}$$

$$\text{s.t.} \quad \text{Constraints (C1), (C2) and (C3).} \tag{12b}$$

We can write problem (12) above using a formal integer programming formulation. However, we do not attempt to solve the problem with a traditional integer programming technique, due to the high computational cost. The generic representation of problem (12) is adequate to understand our computational procedure described in the next section.

Computational Procedure

A procedure to solve problem (12) is to make a series of swaps between the confidential values of different records so that the value of objective function (12a) is reduced after each swap. A swap, which is defined below, must satisfy the constraints in (12b).

> **Definition 3**: *A **swap** of two records refers to an exchange of the confidential values of the two records. That is, a swap between record* $(\mathbf{x}_a, y_a)$ *and record* $(\mathbf{x}_b, y_b)$, *where* $y_a \neq y_b$, *sets the first record to* $(\mathbf{x}_a, y_b)$ *and the second to* $(\mathbf{x}_b, y_a)$.

Clearly, a swap defined this way always preserves the marginal distribution of **y** [constraint (C3)]. Further, we define an admissible swap as follows:

> **Definition 4**: *A swap is **admissible** if constraints (C1) and (C2) remain satisfied after the swap.*

We have developed a set of rules for determining whether a swap is admissible. These rules are closely related to the implementation of the computational algorithm and are discussed in Appendix A. Our algorithm uses a measure, called *cost*, defined below:

> **Definition 5**: *The **cost** of a swap between records i and j is defined as*
>
> $$c_{ij} = [d_i(t_i, r_i) - d_i(t_i, h_i)] + [d_j(t_j, r_j) - d_j(t_j, h_j)], \tag{13}$$
>
> *or equivalently [by substituting equation (11) into (13)]*
>
> $$c_{ij} = d_i(h_i, r_i) + d_j(h_j, r_j), \tag{14}$$
>
> *where t refers to the original confidential value; h and r refer to confidential values before and after the swap, respectively.*

Note that the confidential value for a record before a swap may be different from that record's original confidential value as the record may have been involved in some prior swap. In equation (13), $d_i(t_i, h_i)$ represents the difference in the posterior

probability for the confidential attribute value for record *i* before the swap as compared to the original value, while $d_i(t_i, r_i)$ represents the difference after the swap. If $d_i(t_i, r_i) < d_i(t_i, h_i)$, then the probability of the value after the swap is closer to that of the original value than that before the swap. So, perturbing record *i* from h_i to r_i helps minimize the objective value in problem (12) if $d_i(t_i, r_i) - d_i(t_i, h_i)$ is negative. A similar observation is made for perturbing record *j*. Therefore, a swap involving records *i* and *j* will cause the objective value in problem (12) to decrease if the cost in equation (13) is negative. The computational algorithm for the whole problem is outlined in Table 2. The rules for admissible swaps are presented in Appendix A.

Table 2. Computational Algorithm

1. Find an initial perturbed set by solving problem (1) using linear programmimg.
2. For each record in **U** and **V**, compute posterior probabilities $p(y_{ik} \mid \mathbf{x}_i)(k = 1, ..., C)$ based on equation (10).
3. Compute the cost associated with each admissible swap based on equation (14). Sort those swaps with a negative cost in ascending order. Perform swapping for each of them in the sorted order. After each swap, reset the costs of the remaining swaps that involve any of the two current records to zero, since the admissibility of those swaps needs to be re-examined.
4. Repeat step 3 until all costs are nonnegative.

Table 3. Computational Results for the Example

No.	$p(\text{Low} \mid \mathbf{x})$	$p(\text{Med} \mid \mathbf{x})$	$p(\text{High} \mid \mathbf{x})$	**Original Amount**	**Phase I Perturbed**	$d(t,r)$	**Phase II Perturbed**	$d(t,r)$
1	0.2269	0.7563	0.0168	Med	High	0.7395	Med	0
6	0.2842	0.1895	0.5263	High		0	Low	0.2421
7	0.1698	0.7547	0.0755	Med		0	High	0.6792
10	0.0476	0.6349	0.3175	Low		0	Med	–0.5873
13	0.6090	0.0902	0.3008	Low	Med	0.5188	Low	0
16	0.1130	0.0502	0.8368	High	Low	0.7238	High	0
2	0.7431	0.1651	0.0917	Low	Med	0.5780	Med	0.5780
4	0.0826	0.8257	0.0917	Med	High	0.7339	High	0.7339
11	0.0769	0.0684	0.8547	High	Low	0.7778	Low	0.7778
Sum						**4.0718**		**2.4237**

Next, we discuss the computational complexity of the algorithm. The LP problem in step 1 involves $2C + 1$ constraints, $C(C-1)$ decision variables, and $2C$ slack and surplus variables, where C is the number of confidential categories, which is unlikely to be large in practice. Computing posterior probabilities in step 2 is of order $O(NJC)$, , which is approximately linear in N since N is dominantly larger than C and J (number of attributes). Steps 3 and 4 of the algorithm use a repeated sorting procedure to efficiently perform an exhaustive search for optimal swaps. Since there are at most $N(N-1)/2$ available swaps, the algorithm is guaranteed to converge to zero. Computing the costs in step 3 is of order $O(N^2)$ and sorting the costs is of $O(M \log M)$, where M is the number of swaps having negative cost. Performing swaps on the sorted list and resetting the related costs is of $O(M \log M)$ as well. So, the time complexity for step 3 is of $O(N^2 + M \log M)$. Since there are at most $N(N-1)/2$ total available swaps and step 3 evaluates (and eliminates) $N/2$ swaps, it will take at most $N-1$ loops of step 3 to converge. Therefore, the time complexity for the Phase II swapping procedure is of order $O(N^3 + NM \log M)$. Note that the actual time complexity is substantially lower because there are a large number of inadmissible swaps.

The results of applying the algorithm to the life insurance example are shown in Table 3 (for identifiable records only), where $p = 0.5$, and $C = 3$. Calculations for posterior probabilities in columns 2, 3, and 4 are illustrated in Appendix B. The "Phase I Perturbed" column shows the perturbation results based on the LP solution described earlier. As shown earlier, the marginal distribution of the confidential attribute (Amount) remains the same. The Phase II swaps are shown with arrows. For example, the Amount value for record #1 is perturbed from Med to High in Phase I. A swap between record #1 and record #7 (which is not perturbed in Phase I) is made in Phase II, which resets record #1's value to Med and perturbs the value for record #7 from Med to High. The objective function value, the sum of d_i's, is reduced from 4.0718 to 2.4237 after such swaps.

Experimental Evaluation

We evaluate the effectiveness of the proposed method based on its performance in classification analysis, which is one of the most important data mining tasks. We ran the C4.5 decision tree system (Quinlan 1993) on two real world datasets, described below, and evaluated the proposed method based on classification results such as error rate and tree size. The idea is that if a perturbation method is good, then the classification results based on the perturbed data should be close to those based on the original data.

The Association for Information Systems maintains a Web site that conducts annual surveys of MIS faculty salary offers (Galletta 2004). In this study, we selected the salary offer data in 1999, 2000, 2001, and 2002 (attributes are consistent for these four years and somewhat different for the other years). The dataset consists of 509 records of MIS faculty members who received offers in the period. There are 13 attributes, including salary offered, position, course load, number of years teaching, campus type, public or private, region, year indicator, and so on. The identity-related attributes such as name and e-mail address, if submitted, are deleted from the published data. Salary was considered as the confidential attribute and was chosen as the class attribute. Initially numeric salary values were grouped into three categories. After running a sorting algorithm on the data, we found that 502 out of 509 records are identifiable and 478 of them are uniquely identifiable. So, it is not difficult to discover the salary of a survey participant.

The second dataset, collected from Blake and Merz (1998), was originally extracted from the U.S. Census Bureau databases. It contains 48,842 records, each with 15 attributes (6 numeric and 9 categorical). These attributes provide an individual's demographic information such as age, gender, race, education, occupation, marital status, income, and so on. The income attribute was considered confidential and also chosen as the class attribute. This attribute has two categories, ≤50K and >50K, but the latter

Table 4. Experimental Results

Dataset	Method	Error Rate (%)	Number of Leaves	Perturbation Time (sec.)
Offer	Original	30.84	52	
	Proposed	33.79	75	0.41
	Random	61.12	106	0.31
Census	Original	19.01	507	
	Proposed	30.44	843	38.67
	Random	50.10	2181	6.43

category accounts for less than 25 percent of the total records. This highly unbalanced distribution would either force us to choose a rather small perturbation proportion p or, if p is large, cause the perturbed marginal distribution to significantly depart from the original distribution (also see our earlier discussion on binary confidential attribute). In either case, it will be difficult to evaluate the proposed method. Therefore, we randomly deleted a part of records having the income value of <50K. As a result, the total number of records in the working dataset is 25,049. In order to apply the proposed algorithm, the numeric attributes in the dataset were converted to categorical ones based on the equal-frequency binning method implemented in the Weka data mining package (Witten and Frank 2000). Using the sorting algorithm, we found that 18,278 out of 25,049 records are identifiable and 11,470 of them are uniquely identifiable.

The problem raised in this study has not been investigated previously, so there is no existing data perturbation method against which we can compare our method. We wrote a random perturbation program, which perturbs the confidential values without considering the joint distribution of the data. This random program does attempt to assign confidential values approximately proportional to the original marginal distribution since it is not difficult to do so. For each dataset, we set the perturbation proportion to 0.5 and generated two perturbed datasets, one by the proposed method and the other by the random method, respectively. We then ran C4.5 on each dataset, as well as the original dataset. A 10-fold cross-validation test was conducted for each dataset and the results are shown in Table 4. The perturbation time in the last column refers to the time taken to generate the perturbed data.

It is quite clear that the data generated by the proposed method produced better results than the one by the random method: the error rates and tree sizes based on the data with the proposed method are closer to those on the original data. It is also observed that the proposed method works better on the Offer data than on the Census data. Note that there are three categories in the confidential attribute in Offer but only two categories in Census. When the confidential attribute has only two categories (binary), the proposed method has only the choice of which record to perturb from, but not the choice of which value to perturb to. This may be an explanation for the difference in performance between the two datasets. These observations are, of course, based on two datasets only. In order to make more definitive conclusions regarding the effectiveness of the proposed method, more empirical studies would be desired.

Conclusions and Extensions

We have investigated the privacy protection problem that arises when identifiable records exist in a dataset. We have proposed a two-phase data perturbation approach to the problem. The results of the classification experiments show that the proposed method significantly outperforms the random perturbation method. In future, we plan to conduct experiments using data from different application domains. We also plan to evaluate the proposed method with different data mining tasks, such as clustering and association rules mining.

The proposed method applies to situations where confidential attributes are categorical. In this study, we assume that the confidential attributes are inherently categorical. This is the case in many medical and health management applications (e.g., medical treatment, genetic records, etc.), social science applications (e.g., sexual orientation, academic transcripts, criminal records, etc.), as well as business and financial applications where the grouping of originally numeric attributes is predetermined (e.g., income level specified from multiple choices). In other applications, confidential attributes may be presented in numeric forms. How the values of numeric attributes are grouped will clearly have an impact on the proportion of records in a dataset that can be uniquely identified. This is a related problem that desires further investigation.

In this study, we assume there is only one confidential attribute in the data. When multiple confidential attributes exist, the proposed method can be applied by running the computational procedure multiple times, each run perturbing one confidential attribute in turn. This implementation should be as effective as in the single-confidential-attribute case if there are no or few correlations between the confidential attributes. When such correlations are significant, the proposed method may or may not work well. This is another issue worth further study (see Sarathy and Muralidhar 2002).

References

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

Agrawal, R., and Srikant, R. "Privacy-Preserving Data Mining," in *Proceedings of 2000 ACM SIGMOD International Conference on Management of Data*, Dallas, TX, 2000, pp. 439-450.

Atallah, M., Bertino, E., Elmagarmid, A., Ibrahim, M., and Verykios, V. "Disclosure Limitation of Sensitive Rules," in *Proceedings of the IEEE Knowledge and Data Engineering Exchange Workshop (KEDX'99)*, Chicago, IL, 1999, pp. 45-52.

Blake, C., and Merz. C. J. "UCI Repository of Machine Learning Databases," Working Paper, Department of Information and Computer Science, University of California, Irvine, CA, 1998 (available online at **http://www.ics.uci.edu/~mlearn/MLRepository.html**).

Chowdhury, D. S., Duncan, G. T., Krishnan, R., Roehrig, S. F., and Mukherjee, S. "Disclosure Detection in Multivariate Categorical Databases: Auditing Conidentiality Protection through Two New Matrix Operators," *Management Science* (45:12), 1999, pp. 1710-1723.

Cox, L. H. "Network Models for Complementary Cell Suppression," *Journal of the American Statistical Association* (90:432), 1995, pp. 1453-1462.

Culnan, M. "'How Did They Get My Name?': An Exploratory Investigation of Consumer Attitudes toward Secondary Information Use," *MIS Quarterly* (17:3), 1993, pp. 341-363.

Duncan, G. T., and Mukherjee, S. "Optimal Disclosure Limitation Strategy in Statistical Databases: Deterring Tracker Attacks through Additive Noise," *Journal of the American Statistical Association* (95:451), 2000, pp. 720-729.

Estivill-Castro, V., and Brankovic, L. "Data Swapping: Balancing Privacy against Precision in Mining for Logic Rules," *Data Warehousing and Knowledge Discovery (DaWak'99)*, M. Mukesh and A. M. Tjoa (Eds.), Springer-Verlag, Berlin, 1999, pp. 389-398.

Evfimievski, A., Srikant, R., Agrawal, R., and Gehrke, J. "Privacy Preserving Mining of Association Rules," in *Proceedings of* 8th *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Edmonton, Canada, 2002, pp. 217-228.

Fienberg, S. E., Makov, U. E., and Steele, R. J. "Disclosure Limitation Using Perturbation and Related Methods for Categorical Data," *Journal of Official Statistics* (14:4), 2000, pp. 485-502.

Galletta, D. "MIS Faculty Salary Survey Results," available online at **http://www.pitt.edu/~galletta/salsurv.html**; accessed March 2004.

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

Greengard, S. "Privacy: Entitlement or Illusion?," *Personnel Journal* (75:5), 1996, 74-88.

Kullback, S. *Information Theory and Statistics*, John Wiley & Sons, New York, 1959.

Muralidhar, K., Parsa, R., and Sarathy, R. "A General Additive Data Perturbation Method for Database Security," *Management Science* (45:10), 1999, pp. 1399-1415.

Quinlan, J. R. *C4.5: Programs for Machine Learning*, Morgan Kaufmann, San Mateo, CA, 1993.

Rotenberg, M. "Protecting Privacy," *Communications of the ACM* (35:4), 1992, p. 164.

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

Stanford Student Computer and Network Privacy Project. "A Study of Student Privacy Issues at Stanford University," *Communications of the ACM* (45:3), 2002, pp. 23-25.

Verykios, V. S., Elmagarmid, A. K., Bertino, E., Saygin, Y., and Dasseni, E. "Association Rule Hiding," *IEEE Transactions on Knowledge and Data Engineering* (16:4), 2004, pp. 434-447.

Wang, H., Lee, M. K. O., and Wang, C. "Consumer Privacy Concerns about Internet Marketing," *Communications of the ACM* (41:3), 1998, pp. 63-70.

Witten, I. H., and Frank, E. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, Morgan Kaufmann, San Francisco, CA, 2000.

Appendix A. Rules for Admissible Swaps

Let $\mathbf{U}^p$ and $\mathbf{U}^u$ be the set of perturbed and unperturbed records in $\mathbf{U}$, respectively; and let $\mathbf{V}^p$ be the set of perturbed records in $\mathbf{V}$. For the life insurance example in Table 3, $\mathbf{U}^p$ = {#1,#13,#16], $\mathbf{U}^u$ = {#6,#7,#10} , and $\mathbf{V}^p$ = {#2,#4,#11} . We do not consider the set of unperturbed records in $\mathbf{V}$ because, based on constraint (C2), the confidential value of any record in this set should remain the same in the perturbation process. To describe the set of rules for admissible swaps, we first define a term:

> **Definition 6**: *A **cross-category swap** is a swap between two records that have different original confidential values.*

The essential property of the admissible swap is that perturbation proportions for sets **U** and **V** are unchanged after such a swap. Whether the perturbation proportions will change depends on whether the swap is cross-category, and to which set of $\mathbf{U}^p$, $\mathbf{U}^u$, and $\mathbf{V}^p$ the two records involved in the swap belong. We enumerated all possible scenarios and identified the following rules:

Rule 1: *No swap between two records in* $\mathbf{U}^u$ *is allowed*. This is obvious since the swap will cause $|\mathbf{U}^u|$ to decrease and $|\mathbf{U}^p|$ to increase.

Rule 2: *No swap between* $\mathbf{V}^p$ *and* $\mathbf{U}^u$ *is allowed*. If we allow such a swap, then an unperturbed record in $\mathbf{U}^u$ will be perturbed. Because $\mathbf{U}^p$ is not involved in the swap, the ratio $\frac{|\mathbf{U}^p|}{|\mathbf{U}^u|}$ will change. In the life insurance example, if a swap between records #2 (in $\mathbf{V}^p$) and #6 (in $\mathbf{U}^u$) is made, then set **U** will have four records perturbed, while there are only three perturbed records before the swap.

Rule 3: *No cross-category swap between a record in* $\mathbf{U}^p$ *and a record in* $\mathbf{U}^u$ *is allowed*. To explain this rule, let *a* and *b* be the true and current categories for the record in $\mathbf{U}^p$, respectively. Let *c* be the (true and current) category for the record in $\mathbf{U}^u$. We have $a \neq b$ since the record in $\mathbf{U}^p$ is a perturbed one, $a \neq c$ since this is a cross-category swap, and $b \neq c$ by Definition 3. If this swap is made, then the first record will have a perturbed value of *c*, and the second record will have a perturbed value of *b*. Since the first record is still a perturbed one but the second record is changed from unperturbed to perturbed, $|\mathbf{U}^p|$ is increased by one and $|\mathbf{U}^u|$ is decreased by one.

Rule 4: *A cross-category swap can be made only if the four involved categories (i.e., the first record's true and current categories, and the second record's true and current categories) belong to four different categories*. Let *a* and *b* be the first record's true and current categories, respectively; and let *c* and *d* be the second record's true and current categories, respectively. We have $a \neq c$ since this is a cross-category swap, and $b \neq d$ by Definition 3. If both $a = b$ and $c = d$, then this is the case described in Rule 1; if $a = b$ or $c = d$ but not both, then this is the case described in Rule 3. Therefore, $a \neq b$ and $c \neq d$. If $a = d$, then the first record's confidential attribute is reset to its true value after the swap, which will cause either $|\mathbf{U}^p|$ or $|\mathbf{V}^p|$ to decrease. Therefore, $a \neq d$. Similarly, $c \neq b$. We have established that *a*, *b*, *c*, and *d* must be mutually different.

Appendix B. Calculations of Probabilities in Table 3

We explain here how the posterior probabilities in columns 2, 3, and 4 of Table 3 are calculated. Let us take record #1 as an example. We first compute the prior (marginal) probabilities for the three confidential categories:

P(Low) = 4/16, P(Med) = 6/16, and P(High) = 6/16.

We then compute the conditional probabilities for {Age = 30–39}, given a certain Amount value:

P(30 – 39 | Low) = 2/4, P(30 – 39| Med) = 3/16, and P(30 – 39 |High) = 1/6.

Similarly, we can obtain the conditionals for {Gender = Female} and {Location = CA}, given an Amount value:

P(Female | Low) = 3/4, P(Female | Med) = 3/6, and P(Female | High) = 1/6;
P(CA Low) = 1/4, P(CA | Med) = 5/6, and P(CA | High) = 1/6.

Finally, we compute the posterior probabilities based on equation (10):

$$P(\text{Low} \mid 30-39, \text{Female}, \text{CA}) = \frac{1}{P}(4/16)(2/4)(3/4)(1/4) = (3/128)/P,$$

$$P(\text{Med} \mid 30-39, \text{Female}, \text{CA}) = \frac{1}{P}(6/16)(3/6)(3/6)(5/6) = (5/64)P, \text{ and}$$

$$P(\text{High} \mid 30 - 39, \text{Female}, \text{CA}) = \frac{1}{P}(6/16/)(1/6)(1/6)(1/6) = (1/576)/P,$$

where the normalizing factor $P = P(30 - 39, \text{Female}, \text{CA})$ is not calculated because it will be cancelled out when we normalize the posteriors as follows:

$$P(\text{Low} \mid 30\text{-}39, \text{Female}, \text{CA}) = \frac{(3/128)P}{(3/128)P + (5/64)P + (1/576)P} = 0.2269,$$

$$P(\text{Med} \mid 30\text{-}39, \text{Female}, \text{CA}) = \frac{(5/64)P}{(3/128)P + (5/64)P + (1/576)P} = 0.7563, \text{ and}$$

$$P(\text{High} \mid 30\text{-}39, \text{Female}, \text{CA}) = \frac{(1/576)P}{(3/128)P + (5/64)P + (1/576)P} = 0.0168.$$