Deriving Private Information from Association Rule Mining Results

Zutao Zhu, Guan Wang, and Wenliang Du

Department of Electrical Engineering and Computer Science
Syracuse University, Syracuse, NY, USA 13244
{zuzhu, gwang07, wedu}@syr.edu

***Abstract*— Data publishing can provide enormous benefits to the society. However, due to privacy concerns, data cannot be published in their original forms. Two types of data publishing can address the privacy issue: one is to publish the sanitized version of the original data, and the other is to publish the aggregate information from the original data, such as data mining results. There have been extensive studies to understand the privacy consequence in the first approach, but there is not much investigation on the privacy consequence of publishing data mining results, although, it is well believed that publishing data mining results can lead to the disclosure of private information.**

We propose a systematic method to study the privacy consequence of data mining results. Based on a well-established theory, the principle of maximum entropy, we have developed a method to precisely quantify the privacy risk when data mining results are published. We take the association rule mining as an example in this paper, and demonstrate how we quantify the privacy risk based on the published association rules. We have conducted experiments to evaluate the effectiveness and performance of our method. We have drawn several interesting observations from our experiments.

I. Introduction

Unprecedented amounts of data are being collected on individuals by government agencies, organizations, and industries. If these data can be shared or disseminated, they can bring tremendous benefits to the society, such as improving efficiency of government agencies, enabling us to identify potential pandemic diseases, providing invaluable data for scientific researches, and helping commercial industries to gain a better understanding of the market and customers. For these reasons, an increasing amounts of data are being disseminated.

Many of these data contain personal information, and a daunting challenge faced by data publishing is how to protect privacy, i.e., how to keep individual's private information from being disclosed. Such privacy concerns result in two lines of approaches. One approach focuses on sanitizing the original datasets before publishing them. Many sanitization methods have been proposed, including randomization [1]–[4], generalization [5]–[9], and bucketization [10].

Another way to publish data without disclosing too much private information is to only publish data mining results, instead of datasets. Because data mining results contain only aggregate information, they can achieve better privacy protection; moreover, since data mining results are generated from the original data, they are more accurate than those from sanitized data. The downside of this approach is that users cannot apply their own data mining or analysis methods on the data; they have to take whatever is produced by the data publishers.

The privacy consequence of the first approach has been extensively studied in the literature [10]–[16]; however, there is not much study on the privacy consequence of the second approach. Although publishing mining results seems to be able to achieve a better privacy protection than publishing datasets, it is still widely believed that data mining results do contain quite a significant amounts of information that has the potential to compromise privacy. Several work has studied the privacy consequence when frequent itemsets are published [17]–[19]. The approach is to reconstruct a dataset from the published frequent itemsets, and then use the reconstructed dataset to analyze privacy. This field is called *inverse frequent itemset mining*. Another related work is carried out by Kantarcioglu et al., who try to answer when data mining results violate privacy [20]. However, despite all these studies, there is a lack of understanding on how much private information can be disclosed by data mining results.

The objective of this paper is to develop a systematic method to quantify privacy disclosure caused by the publishing of data mining results.

A. Motivation and Challenges

In the field of privacy-preserving data publishing, it is a convention to conceptually split datasets into two parts: Quasi-identifier (QI) attributes and Sensitive Attributes (SA). The QI part consists of the information that are not sensitive and can usually be obtained from other sources; the SA part consists of private information of individuals. The goal of privacy-preserving data publishing is to prevent adversaries from inferring any individual's SA information, while making the published information as useful as possible.

However, when some information about the original data is published, such as sanitized data or data mining results, adversaries might be able to infer an individual's SA information based on the published information and the QI information of the individuals that they have obtained from other sources. This type of attacks is often referred to as *linking attack*, i.e., linking an individual's QI to SA value [9]. The severity of linking attacks is decided by the conditional probability

ID	Education	Gender	Salary
1	Doctorate	Male	$50K-$
2	Masters	Female	$50K-$
3	Doctorate	Female	$50K+$
4	Bachelors	Male	$50K-$
5	Masters	Female	$50K+$
6	Doctorate	Male	$50K+$
7	Masters	Female	$50K+$
8	Doctorate	Female	$50K+$
9	Masters	Female	$50K+$
10	Doctorate	Female	$50K+$
11	Masters	Female	$50K+$
12	Doctorate	Female	$50K+$

(a) The data set D ("50K+" means the salary is $> 50K$, and "50K-" means the salary is $\leq 50K$)

Index	Association Rule
	Minimum Support: 0.3, Minimum Confidence: 0.8
1	$Education = \texttt{Doctorate} \implies Salary = \texttt{50K+}$ confidence: (0.83), support: (0.42)
2	$Education = \texttt{Doctorate} \wedge Gender = \texttt{Female} \implies Salary = \texttt{50K+}$ confidence: (1), support: (0.33)
3	$Gender = \texttt{Female} \implies Salary = \texttt{50K+}$ confidence: (0.89), support: (0.67)

(b) Data mining results: association rules

Fig. 1. Dataset and association rules for Example 1.

$P(SA \mid QI)$ derived from the published data. The closer this probability is to 1, the more certain adversaries can infer the SA value of an individual with QI. To quantitatively understand the privacy consequence of data publishing, we need to put ourselves at the adversaries' shoes and derive $P(SA \mid QI)$. Unfortunately, this is not an easy task. We use two examples to illustrate this in the association-rule mining scenario.

Example 1: Figure 1 depicts a dataset and the association rules generated from it with support threshold 0.3 and confidence threshold 0.8. In the dataset, *Salary* is the only SA attribute. We assume that adversaries already know the QI part of the data (they can get the information from other sources); we also assume that adversaries know the domain of *Salary* is {`50K+, 50K-`}. Therefore, the useful association rules that can help adversaries derive each individual's SA value are those of pattern $QI \implies SA$ (i.e., the right hand of the rules consists of only sensitive attributes). We only focus on this pattern.

If the association rules in Figure 1(b) are published, we can directly derive $P(SA \mid QI)$ and $P(QI, SA)$ for certain QI and SA values. For example, from the third association rule, we can get the following:

$$P(Gender = \texttt{Female}, Salary = \texttt{50K+}) = 0.67,$$
$$P(Salary = \texttt{50K+} \mid Gender = \texttt{Female}) = 0.89.$$

Even if the exact confidence and support of each rule is suppressed from the disclosure, we can still derive the following inequalities:

$$P(Gender = \texttt{Female}, Salary = \texttt{50K+}) \geq 0.3.$$
$$P(Salary = \texttt{50K+} \mid Gender = \texttt{Female}) \geq 0.8,$$

The above probability derivations are based on a single association rule; adversaries can even combine the information from several rules to make more derivations. For example, from the first rule in Figure 1(b), we know that the percentage (P_d) of doctorates who earn more than 50K is 83%; from the second rule, we know that 100% of female doctorates earn more than 50K. Putting these two rules together, we know that the percentage (P_{md}) of male doctorates who earn more than 50K is less than 83%. If we factor in the number of male doctorates (2 people) and female doctorates (4 people), we can even say that only one of the two male doctorates earn more than 50K (i.e. $P_{md} = 0.5$): if both had earned more than 50K, P_d would have been 100%.

The above example demonstrates that adversaries can gain more accurate knowledge if they combine the information from several rules. Obviously, we cannot use a combinatorial approach to try all possible combinations, because the number of combinations, which is exponential in the number of association rules, can be too many.

To make the matter worse, it is not sufficient to only consider the published association rules; the fact that a pattern $QI \implies SA$ is not an association rule also gives adversaries useful information. We demonstrate this using the following example.

Example 2: Figure 2 gives a simple dataset (the original dataset). The owner of the dataset has published the following association rule using 0.6 as the support threshold and 0.9 as the confidence threshold:

$$Education = \texttt{Doctorate} \implies Salary = \texttt{50K+},$$
$$support = 0.67, \quad confidence = 1.$$

ID	Education	Gender	Salary
1	Doctorate	Female	$50K+$
2	Doctorate	Male	$50K+$
3	Masters	Female	$50K-$

Fig. 2. Dataset for Example 2.

Suppose that adversaries know the QI part of the data set and the domain of salary as Example 1. From the published association rule, adversaries can immediately learn that the salaries for the first and second records are "`50K+`". However, that is not all. Adversaries can derive more from the things that are not published.

Knowing that the pattern "$Gender = \texttt{Female} \implies Salary = \texttt{50K+}$" is not published as an association rule, adversaries can conduct the following reasoning (we denote the pattern as T): If the third person's salary were `50K+`, the support of T would be 0.67 and the confidence would be 1 (note: from the published association rule, adversaries have already known that the first record's SA value is `50K+`); therefore, the pattern T would have satisfied both support and confidence thresholds and should have been published as an association rule. This contradicts to the fact, so clearly, the third person's salary is `50K-`.

For a large data set, many patterns do not become association rules; trying to combine them with the published

association rules to derive useful information is computationally infeasible. We need a more systematic way to combine the information embedded in both published association rules and unqualified patterns, and use the combined information to derive $P(SA \mid QI)$.

B. Outline of Our Approach

We have developed a generic and systematic method to derive $P(X \mid Q)$ from data mining results (we use X to denote SA values and Q to denote QI values) . We formulate the derivation of $P(X \mid Q)$ as a non-linear programming problem. We treat $P(X \mid Q)$ as a variable for each combination of $X \in SA$ and $Q \in QI$. The goal of deriving $P(X \mid Q)$ is to assign probability values to these variables. As we have said before, data mining results contain information about $P(X \mid Q)$, so the assignment of these probability variables should be consistent with the information embedded in the data mining results. Usually, the embedded information can be formulated as constraints, which are in the forms of equations or inequalities. Therefore, the derivation of $P(X \mid Q)$ becomes finding an assignment for these variables that satisfy the constraints.

Many assignments might be possible (i.e., they are all consistent with the published mining results); the question is which one should be used. To understand our decision, we need to go back to the actual meanings of the variables: these variables are not abstract variables; they are probabilities, so assigning values to these variables is actually called probability estimate. It is well known that when conducting estimates, it is desirable to be as unbiased as possible.

According to the *principle of Maximum Entropy (ME), when the entropy of these variables is maximized, the inference is the most unbiased* [21]. Therefore, applying the ME principle, our problem becomes finding the maximum-entropy assignment for those variables that satisfy the constraints formulated from the published data mining results. The ME problem is a special case of nonlinear programming problems, and many existing numeric methods [22], [23] and software packages [24], [25] can be used to solve this type of problem in a systematic way.

Our method can be applied to a variety of data mining results, including association rules, classifiers (e.g., decision trees, association-rule based classifiers), etc., as long as we can derive the constraints from the information embedded in the data mining results. In the scope of this paper, we focus on demonstrating how to use our proposed method on association-rule mining results.

C. Paper Organization

In Section II, we summarize related work. In Section III, we formally formulate the privacy quantification as probability estimation problem. In Section IV, we briefly describe the Principle of Maximum Entropy, which is used as the theoretic foundation of our method. In Section V, we describe how to derive constraints from the published association rules. In Section VI, we use experiments to study the effectiveness and performance of our method. Finally, we conclude in Section VII.

II. Related Work

The general scope of the related studies is called privacy-preserving data publishing, the goal of which is to publish data without compromising privacy requirements. The majority of the work has been focusing on how to sanitize data, so they can be released without causing privacy breaches. A number of data sanitization methods have been proposed, including randomization [1]–[4], generalization [5]–[9], and bucketization [10], [15]. For these methods, adversaries have a sanitized version of the original data. Our work focuses on a different type of data publishing, where only data mining results are published, not datasets.

The effect of data mining results on privacy was studied by Kantarcioglu et al. [20]. In this work, a classifier is modeled as a "black box", namely, adversaries can only request that an instance be classified by the owner of a classifier; they can get the classification results, but can get no other information about the classifier. Although this model has its own merit in the client/server model, where mining results are kept at a sever, it is an unrealistic model for the scenario where data mining results are indeed published. Our work treats data mining results as "white boxes", i.e., the mining results are fully accessible to adversaries.

The privacy consequence of frequent itemset mining results has been studied in [17]–[19]. Results of frequent itemset mining can compromise privacy in several ways. First, some frequent itemsets can contain private information, and should not be published; although non-sensitive frequent itemsets do not contain private information directly, they may be used to derive information about the sensitive frequent itemsets. This is called *inference channels* between sensitive and non-sensitive itemsets in [18]. In this paper, Wang et al. give some algorithms to block the inference channels using itemset sanitization. Second, frequent itemset mining results can be used to construct a synthetic dataset that is consistent with the original data [17], so adversaries can analyze the features of original data using the synthetic data. In [19], Wang et al. analyze the privacy consequence of such synthetic data construction. They propose an approximate construction method. The above studies are called *inverse frequent itemset mining*; they focus mainly on developing algorithms to construct the synthetic dataset using the frequent itemset mining results, or on publishing these frequent itemsets without compromising privacy. There is a lack of study on understanding how much privacy is compromised due to the published frequent itemsets. Our paper tries to address this privacy measuring problem.

Applying the principle of maximum entropy to understand privacy in privacy-preserving data publishing was first explored in our work [26], which focuses on understanding how background knowledge affects the privacy if sanitized data are published. Although also based on the principle of maximum entropy, this paper is significantly different from the work in [26]. First, while the work in [26] focus on publishing

sanitized data, this paper focuses on another data publishing approach, i.e., publishing data mining results. Therefore, the information available to adversaries is different. Second, this paper makes an interesting hypothesis, which states that not only the published association rules contain private information, the fact that certain patterns are not published as association rules also contains a great deal of information. This paper proposes an efficient algorithm to derive constraints from all the information available to adversaries. On the other hand, because this work and the work in [26] are both based on the principle of maximum entropy, they can be combined quite naturally to quantify privacy caused by the publishing of data mining results, while assuming that adversaries know a certain degree of background knowledge. All we need to do is to put the constraints from both sources together, and feed them into a nonlinear programming solver. Due to page limitation, we will not pursue such an integration work in this paper.

Another line of research that is quite relevant to our studies is the derivation of private information based on the algorithms that are used in data disguise. In [27], Wong et al. found that all former anonymization algorithms try to minimize information loss during data disguising. This fact on optimization actually provides useful information to adversaries. Attacks based on such facts are called *minimality attack* in [27]. Similar attacks are also described in [28]. These attacks are based on the information that is not published, but is implied from the published information. For example, in the minimality attack, the fact that a specific version of sanitized data is not published tells us that the version cannot achieve the optimal results. Our work explores a similar types of information: the fact that a pattern is not an association rule indicates that the pattern fails to satisfy the minimum confidence or support thresholds.

III. Problem Formulation

A. Assumptions

We make several assumptions in this paper.

- The original dataset consists of two parts: QI attributes and SA attributes. The QI part consists of the information that can also be obtained from other sources. The SA part consists of the information that the data owner wants to protect.
- For the sake of simplicity in this paper, we assume that there is one SA attribute in the data set. However, our method is not restricted to this assumption, it can be easily extended to datasets that have multiple SA attributes [1].
- We assume that adversaries have all the data of the QI attributes. This assumption is made because the information in the QI part can be usually obtained via other means [9]. Although in practice, attackers might not know every QI value, this assumption allows us to conduct analysis on the worse-case scenario.
- We assume that adversaries know the domain of the sensitive attributes, i.e., they know all the possible values of the sensitive attributes.
- We focus only on categorical data in this paper, because association rule mining is mostly based on categorical data.

[1]If there are multiple categorical sensitive attributes, we can enumerate all the SA combinations. We assign a unique ID for each combination so that it can be treated as a single SA.

B. Measuring Privacy

Quantifying privacy has been actively pursued by researchers in the past few years. Several privacy metrics have been proposed, including K-anonymity [9], L-diversity [12], (α, k)-anonymity [29], t-Closeness [13], etc. A common thread behind these metrics can be described as the following: using an individual's QI information acquired from another source, together with the other information provided to the adversaries (such as anonymized datasets), adversaries might be able to derive information about individual's SA value. How successful the adversaries can derive an individual's correct SA value depends on the intrinsic conditional probability between QI and SA attributes, i.e., $P(SA \mid QI, \mathcal{O})$, where $\mathcal{O}$ represents all the information available to the adversaries.

In most of the existing studies, $\mathcal{O}$ consists of the information from sanitized datasets [9], [10], [12], [13], [29]. In our study, we focus on a different type of information that is available to adversaries, i.e., data mining results. For the sake of simplicity, we omit $\mathcal{O}$ from our notation, and only use $P(SA \mid QI)$ in the rest of the paper. Our privacy quantification task can be formally defined as the following:

Problem 3.1: Let D be the original data set that is used to generate the data mining results (denoted as Ω). Let variable X represent SA attributes, and variable Q represent QI attributes. Given Ω and the QI part of all the records in D, derive $P(X \mid Q)$ for all the combinations of Q and X values.

The value of $P(X \mid Q)$ is the primitive behind all the existing privacy measures, i.e., as long as we can compute this conditional probability, we can calculate the existing privacy metrics, such as L-diversity [12], (α, k)-anonymity [29], etc.

IV. Maximum Entropy Modeling

Directly computing $P(X \mid Q)$ is quite complicated, especially when various sources of information have to be considered. Instead of directly computing $P(X \mid Q)$, we treat $P(X \mid Q)$ for each combination of Q and X as a variable; for example, if there are 1000 combinations, we have 1000 variables. Our goal is to assign probability values to these variables, while ensuring that the assignment is consistent with the information encoded in the published data mining results.

It is possible that many assignments of $P(X \mid Q)$ are consistent with a given set of data mining results, with some being biased toward certain particular X values. Being biased means assuming some extra information that we do not possess; therefore, the least biased assignment is the most desirable. As E. T. Jaynes expounded in his theory, the least biased assignment is achieved when the probability entropy is maximized [21].

This is the *Maximum Entropy (ME)* principle. Applying this principle, our problem becomes finding a distribution of $P(X \mid Q)$, such that the following conditional entropy $H(X \mid Q)$ is maximized:

$$H(X \mid Q) = -\sum_{Q,X} P(Q)P(X \mid Q)\log P(X \mid Q). \quad (1)$$

Sometimes, it is more convenient to use the ME method to compute $P(Q, X)$ first [2], then compute $P(X \mid Q)$. In this case, we maximize the following joint entropy $H(Q, X)$:

$$H(Q, X) = -\sum_{Q,X} P(Q, X)\log P(Q, X). \quad (2)$$

Because $H(X \mid Q) = H(Q, X) - H(Q)$, when $H(Q)$ is a constant, maximizing $H(X \mid Q)$ is equivalent to maximizing $H(Q, X)$. Since we have assumed that adversaries know the QI part of the dataset, there is no uncertainty on Q, i.e., $H(Q)$ is a constant. In the rest of this paper, for the sake of simplicity, our discussion is mainly based on $H(Q, X)$.

Without any constraint, $H(Q, X)$ is maximized when $P(Q, X)$ has a uniform distribution. However, the values of $P(Q, X)$ are indeed subject to many constraints contained in the data mining results. To apply the ME method, we need to convert all the available knowledge into equations (or inequalities) based on $P(Q, X)$. Let these constraints be c_1, ..., c_w. Our problem can be formally defined as the following:

Definition 4.1: (Maximum Entropy Modeling) Finding an assignment for $P(Q, X)$ for each combination of Q and X, such that the entropy $H(Q, X)$ is maximized, while all the constraints c_1, ..., c_w are satisfied.

This maximum entropy modeling problem is a special case of the non-linear programming problem, which can be solved using existing software, such as KNITRO [24] and the TOMLAB/SOL toolbox [25].

V. Deriving Private Information from Association Rule Mining Results

Based on the ME method described in the previous section, to estimate $P(X \mid Q)$ based on data mining results, all we need to do is to convert the knowledge embedded in data mining results into equations or inequalities using $P(X \mid Q)$ or $P(Q, X)$ as variables. We call these equations and inequalities ME constraints. Once we have formulated constraints, we can use the existing solutions to find the values of $P(X \mid Q)$, for each instance of Q and X. In this section, we describe how to formulate ME constraints.

The difficulty of deriving constraints from data mining results depends on the type of data mining results and the algorithms used to generate data mining results. According to whether they are related to probabilities, data mining results can be classified into two categories: probability-based and nonprobability-based. For example, association rule mining is probability-based, but most clustering algorithms are not. Our method is suitable for probability-based mining results; it is unclear how to extend it to nonprobability-based mining. In this section, we describe how to formulate constraints from the published association rules.

[2]This is mainly because formulating constraints using $P(Q, X)$ sometimes is much easier than using $P(X \mid Q)$.

A. Deriving Constraints From Association Rules

We briefly review the basics of association rule mining, before discussing how to derive ME constraints from association rules.

Association Rules Mining. An association rule is an expression $S \Longrightarrow T$, where S and T are sets of items. The goal of association rule mining is to find out all the association rules with *support* above a minimum threshold s and *confidence* above a minimum threshold c. For an association rule $S \Longrightarrow T$, the *support* of a rule is defined as the fraction of records that contain both S and T, i.e., support is defined as $P(S \wedge T)$; the *confidence* of a rule is a percentage value that shows how frequently T occurs among all the groups containing S, i.e., confidence is defined as $P(T \mid S)$. One of the most well-known algorithms to generate association rules is the Apriori algorithm [30].

In the context of our studies, data sets contain two types of attributes, QI attributes and SA attributes. For this type of data sets, the association patterns between QI and SA attributes are the most interesting association rules that are worth publishing. Therefore, when we say publishing association rules, we mean publishing the rules of the type $Q \Longrightarrow X$, where Q consists of values of QI attributes and X contains of values of SA attributes.

We use an example to illustrate the association rules concept. Using a portion of the UCI Adult dataset [31], by setting the support threshold $s = 0.1$ and the confidence threshold $c = 0.9$, we can get the following association rule:

$$MaritalStatus = \texttt{NeverMarried},\ Sex = \texttt{Male},$$
$$\Longrightarrow Salary = \texttt{50K}-,$$
$$support = 0.168,\quad confidence = 0.901.$$

This is a qualified association rule because the support value is $0.168 > s$ and the confidence value is $0.901 > c$.

Deriving Constraints. Although association rules only contain aggregate information about original datasets, they have potentials to disclose the private information in the original datasets. To use our ME method to quantify how much private information is disclosed by these data mining results, we need to derive constraints from these rules. These constraints should be in the form of $P(Q, X)$. We call the constraints derived from association rules *AR-constraints*.

There are two potential scenarios when publishing association rules. In the first scenario, data owners withhold the exact support and confidence scores. That is, users (and adversaries) only know the published association rules together with the thresholds for support (s) and confidence (c). In this situation, for an association rule $Q \Longrightarrow X$, we only know the following:

$$P(X \mid Q) \geq c \text{ and } P(Q, X) \geq s.$$

Because our ME constraints need to be based on $P(Q, X)$. We rewrite the first inequality using $P(Q, X) = P(X \mid Q) \cdot P(Q)$; combining with the second inequality, we get the following constraint (it should be noted that adversaries know $P(Q)$ based on our assumption [3]):

$$P(Q, X) \geq max(s, c \cdot P(Q)). \quad (3)$$

In the second scenario, to give users more information about the association rules, data owners also publish the exact support and confidence scores for the association rules. That is, when publishing an association rule $Q \Longrightarrow X$, the confidence score $c' = P(X \mid Q)$ and the support score $s' = P(Q, X)$ are also published. Based on these scores, we can derive the following constraints:

$$P(X \mid Q) = c' \text{ and } P(Q, X) = s'. \quad (4)$$

Since $P(Q, X) = P(X \mid Q) \cdot P(Q)$ and $P(Q)$ is also known, one of the above equations is redundant. We keep the second one as our ME constraint, i.e., $P(Q, X) = s'$.

For each of our ME constraint, the Q part in $P(Q, X)$ must include all the quasi-identifier attributes. We call such an probability expression a *full probability expression*. Unfortunately, in any association rule $Q_s \Longrightarrow X$, Q_s usually does not include all the QI attributes. Using the above derivation, we can only derive AR-constraints based on $P(Q_s, X)$. If this is the case, we can use the following theorem to extend each $P(Q_s, X)$ to a sum of several full probability expressions:

Theorem 1: Let Q represent the set of entire QI attributes and Q_s is a nonempty subset of Q. Let $\bar{Q_s} = Q - Q_s$ be the difference between Q and Q_s. Therefore, $P(Q, X)$ can be written as $P(Q_s, \bar{Q_s}, X)$. By summing up the probability over all possible $\bar{Q_s}$ values, we have the following:

$$\sum_{t \in \bar{Q_s}} P(Q_s, t, X) = P(Q_s, X). \quad (5)$$

B. Deriving Constraints From Non-Association Rules

The published association rules are all the information that is available to users and adversaries. We have used ME constraints to capture the information revealed by these association rules. An important question is whether that is all the information revealed by the published data mining results. The answer is no. We have only captured the knowledge from *positive information*, i.e., the published association rules, but we have not captured the knowledge from *negative information*, i.e., those patterns that are disqualified as association rules.

If a pattern $Q \Longrightarrow X$ is missing from the published association rules, it actually tells us some information about Q and X: this pattern fails to satisfy either the minimum support threshold s or the minimum confidence threshold c. This kind of knowledge can be modeled as ME constraints too. We call this pattern a *non-association rule*, and the constraints derived from it the *NAR-constraints*.

[3]We assume that adversaries know the QI part of the data, i.e., they already know $P(Q)$.

For any combination of Q and X, if $Q \Longrightarrow X$ is not one of the published association rules, we can derive the following constraints:

$$P(X \mid Q) < c \text{ or } P(Q, X) < s.$$

Unfortunately, the Maximum Entropy model cannot accommodate the above two constraints, because the model requires all the constraints to have *and* relationships, i.e., they should all be satisfied. If two constraints have an *or* relationship, we have to split them into two sets of ME constraints, each has to be solved separately. If there are too many or-related constraints, the number of constraint sets will be exponentially large, rendering the problem infeasible to solve.

Fortunately, the above two constraints can be merged into one constraint. Based on the fact that $P(Q, X) = P(X \mid Q) \cdot P(Q)$, we can rewrite the first constraint as $P(Q, X) < c \cdot P(Q)$; combining with the second one, we have a single constraint:

$$P(Q, X) < max(s, c \cdot P(Q)). \quad (6)$$

Since s and c are both constants, the right hand of the above inequality depends on $P(Q)$, which is the knowledge that adversaries already have (by knowing the QI part of the data).

The Number of Constraints. Since any pattern that is not an association rule can lead to a NAR-constraint, plus the AR-constraints, there might be many of constraints. This can cause a problem when we try to solve the nonlinear programming problem based on the constraints, because the number of constraints and the number of variables in these constraints are the two important factors for performance and memory usages.

We study the total number of constraints that might be generated. For the sake of simplicity, we assume that there are m QI attributes, each with k distinct values. We have the following theorem:

Theorem 2: Let $|SA|$ be the total number of distinct SA values. The total number of constraints is the following:

$$|SA| \cdot \sum_{i=1}^{m} k^i \cdot \binom{m}{i} = |SA| \cdot ((1+k)^m - 1). \quad (7)$$

Proof: Let $Q^{(i)}$ be an i-itemset which consists of i QI attributes. Let X be a sensitive attribute assignment. For $Q^{(i)}$, the number of patterns $Q^{(i)} \Longrightarrow X$ is k^i, because each attribute in $Q^{(i)}$ can take k different values.

Since there are $\binom{m}{i}$ different i-itemsets, the total number of patterns $Q \Longrightarrow X$ (association rules or non-association rules) is the following (where Q is any i-itemset):

$$k^i \cdot \binom{m}{i}.$$

Multiplying the above expression with $|SA|$, the number of distinct SA values, and sum it over $i = 1 \ldots m$, we can get Equation (7). ■

For the UCI Adult dataset, which has 30162 records and 8 QI attributes after preprocessing, there are about $700,000$ constraints (including both AR-constraints and NAR-constraints), most of which are actually NAR-constraints because the number of AR-constraints is the same as the number of published association rules, which is small in practice. Many Non-Linear programming solvers can handle this amount of constraints, if they are short, i.e., not containing too many variables. However, this is not true in our case, as there are more than $3,900,000$ variables among these constraints, including the duplicates (some variables occur in several constraints, and they are counted based on their occurrences). An insight in how Non-Linear Programming solvers are implemented tells us that the amount of memories used by the solvers depend on the number of variables. If the number is too large, the solvers can run out of memories. We have indeed tried several solvers (KNITRO [24] and TOMLAB [25]) on this scale, and none of the solvers that we have tried can handle a problem of this scale. Therefore, it is very essential to reduce the number of constraints, so the total number of variables can be reduced.

C. An Efficient Algorithm to Derive NAR-Constraints

We have observed that not all the NAR-constraints are necessary, some of them are actually redundant. A constraint is said to be *redundant* if it can be derived from one or more of other constraints. Let us see an example. Suppose that we have found two patterns $Q \Longrightarrow X$ and $Q \wedge q \Longrightarrow X$ that are not association rules, where q contains another QI attribute that does not exist in Q. We might end up having the following two constraints corresponding to these patterns: $P(Q, X) < s$ and $P(Q \wedge q, X) < s$. The second one is actually redundant, because it can be derived from the first one using the fact that $P(Q \wedge q, X)$ is always less than or equal to $P(Q, X)$.

The above reasoning is used by the Apriori algorithm [30] when generating the itemsets for association rules. It takes advantage of the fact that if $P(Q, X) < s$, then $P(Q \wedge q, X) < s$. In this way, we can exclude $(Q \wedge q, X)$ from the candidate itemset, if (Q, X) is not in the candidate itemset. Our process of generating NAR-constraints is quite similar to the process of generating frequent itemsets. Therefore, it is desirable if we can apply the same technique to prune those redundant NAR-constraints during the constraint generation process.

Unfortunately, we cannot directly apply the same reasoning to generate NAR-constraints. The key difference between our process and the Apriori process is the right hand of the inequalities. Let us still use the patterns $Q \Longrightarrow X$ and $Q \wedge q \Longrightarrow X$ as an example. In the Apriori algorithm, the right hands of $P(Q, X) < s$ and $P(Q \wedge q, X) < s$ are the same number, so the first one implies the second one. In our process, according to Equation (6), our constraints are $P(Q, X) < \max(s, c \cdot P(Q))$ and $P(Q \wedge q, X) < \max(s, c \cdot P(Q \wedge q))$; the right hands of these two inequalities are different numbers, so the first one does not necessarily imply the second one.

However, under certain condition, the second NAR-constraint can be pruned if we already have the first NAR-constraint. We have the following theorem:

Theorem 3: Suppose that we have two NAR-constraints $P(Q, X) < \max(s, c \cdot P(Q))$ and $P(Q \wedge q, X) < \max(s, c \cdot P(Q \wedge q))$. If $c \cdot P(Q) \leq s$, the second constraint is redundant.

Proof: To prove this theorem, we just need to show that the second constraint can be derived from the first one.

From $c \cdot P(Q) \leq s$, we can derive that $c \cdot P(Q \wedge q) \leq s$, because $P(Q \wedge q) \leq P(Q)$. Therefore, these two NAR-constraints can be rewritten as the following:

$$P(Q, X) < s \text{ and } P(Q \wedge q, X) < s.$$

Clearly, the second constraint can be derived from the first one, because $P(Q \wedge q, X) < P(Q, X)$. ■

What Theorem 3 tells us is that if we have derived a NAR-constraint from a pattern $Q \Longrightarrow X$, and if we know $c \cdot P(Q) \leq s$, the NAR-constraint derived from any pattern $Q \wedge Q' \Longrightarrow X$ is redundant, where Q' consists of a set of QI attributes (and their values) that are not in Q. Therefore, all these NAR-constraints can be pruned in the process of constraint generation.

Based on Theorem 3, we can derive an efficient algorithm to generate NAR-constraints (it actually generates both AR- and NAR- constraints). The algorithm is depicted in Figure 3, and it is similar to the Apriori algorithm, but using different pruning criterion. In the algorithm, L_{k-1} is the set of qualified $(k-1)$-itemsets which is to be extended to k-itemset. An itemset I_e is said to be a *one-extension* of another itemset I if $I_e = I \wedge q$ where q consists of a single attribute that is not in I. For example, the 3-itemset $Race = \texttt{White} \wedge Sex = \texttt{Male} \wedge MaritalStatus = \texttt{Divorced}$ is a one-extension of the 2-itemset $Race = \texttt{White} \wedge Sex = \texttt{Male}$.

The generation of NAR-constraints for k-itemset is based on the AR-constraints and NAR-constraints of $(k-1)$-itemset. Initially, we put all the one-itemset into L_1 and generate the AR-constraints and NAR-constraints accordingly. We then iteratively extend the $(k-1)$-itemset L_{k-1} to k-itemset L_k using a procedure similar to the Apriori algorithm, i.e., for each one-extension itemset I of an element in L_{k-1}, if $I \Longrightarrow X$ is not an association rule, we check whether the condition in Theorem 3 is satisfied. If it does, the itemset I will be pruned; otherwise, it will be added to L_k.

The algorithm is quite effective; it reduces the number of NAR-constraints from $700,000$ to less than $1,000$ in our experiments using the UCI adult data set.

D. Deriving Constraints from Quasi-Identifiers

In addition to the constraints derived from the association rules and non-association rules, there is another set of constraints that have not been captured. These constraints are due to the fact that $P(Q, X)$'s are probabilities, and they should satisfy all the constraints imposed on probabilities; a trivial example is that the sum of all these probabilities should be 1. It should be noted that Non-Linear Programming solvers do not recognize the variables in the constraints as probabilities, so it has no responsibility to ensure all the constraints related to probabilities. We have to specifically formulate them.

Input: The QI part of the dataset D_{qi}, threshold s and c
Output: Φ_k, set of AR-constraints for k-itemset,
and Ψ_k, set of NAR-constraints for k-itemset
$L_1 = \{1\text{-itemset}\}$;
// Initialize step;
foreach *itemset* $I \in L_1$ **do**
  for $x \in X$ **do**
    if $I \Rightarrow x$ *is an association rule* **then**
      generate an AR-constraint and put it in Φ_1;
    end
    if $I \Rightarrow x$ *is not an association rule* **then**
      generate a NAR-constraint and put it in Ψ_1;
    end
  end
end
// Iteration step;
for $k \leftarrow 2$ **to** m **do**
  foreach *itemset* $I_{k-1} \in L_{k-1}$ **do**
    foreach I, *one-extension of* I_{k-1} **do**
      for $x \in X$ **do**
        if $I \Rightarrow x$ *is an association rule* **then**
          generate an AR-constraint and put it in Φ_k;
          add I into L_k;
        end
        if $I \Rightarrow x$ *is not an association rule* **then**
          optimize-gen(Ψ_k, Φ_{k-1}, Ψ_{k-1}, I, x, L_{k-1}, s, c);
        end
      end
    end
  end
end

procedure optimize-gen($\Psi_k, \Phi_{k-1}, \Psi_{k-1}, I, x, L_{k-1}, s, c$)
1: **if** $(I_{k-1} \Rightarrow x) \in \Psi_{k-1} \wedge c \cdot P(I_{k-1}) \leq s$ **then**
2: return ($I \Rightarrow x$ is pruned based on Theorem 3)
3: **else**
4: add the NAR-constraint for the pattern $I \Rightarrow x$ into Ψ_k;
5: add I into L_k;
6: **end if**

Fig. 3. An efficient algorithm to generate NAR-constraints.

There are three types of constraints that should be imposed on joint probabilities $P(Q, X)$'s. First, if we sum $P(Q, X)$'s over all possible X values, the result should be $P(Q)$. Let m represent the total number of distinct values for the sensitive attribute, and let $x_1, \ldots, x_m$ be these values. We have the following constraint for each QI value q:

$$\sum_{i=1}^{m} P(Q = q, X = x_i) = P(Q = q), \tag{8}$$

where $P(Q = q)$ is the probability of q in the original data set. Since we assume that the adversaries know the QI part of the data set, $P(Q = q)$ is known to the adversaries. We call the above constraint *QI-constraints*.

Second, similar to the QI-constraints, if we sum $P(Q, X)$'s over all possible Q values, the result should be $P(X)$. Let n represent the total number of distinct values for the QI attributes, and let $q_1, \ldots, q_n$ be these values. We have the following constraint for each SA value x:

$$\sum_{i=1}^{n} P(q_i, X = x) = P(X = x), \tag{9}$$

where $P(X = x)$ is the probability of x in the original data set. We call the above constraint *SA-constraints*. If the distribution of SA values are also published along with the association rules, adversaries will know $P(X = x)$, so the above SA-constraints should be included. However, if the data owners decide to withhold this piece of information, adversaries will not know $P(X = x)$, and these constraints should not be included.

The third type of constraint is that if we sum $P(Q, X)$'s over all the possible QI and SA values, the result should be 1. However, this constraint is redundant, because if we add all the QI-constraints or all the SA-constraints, the result will be the same as this constraint.

We show an example of the QI-constraint here. For the dataset depicted in Figure 1(a), we can get the following QI-constraint. $P(Education = \texttt{Doctorate} \wedge Gender = \texttt{Female}, Salary = \texttt{50K+}) + P(Education = \texttt{Doctorate} \wedge Gender = \texttt{Female}, Salary = \texttt{50K-}) = \frac{4}{12} = \frac{1}{3}$.

VI. Experiments

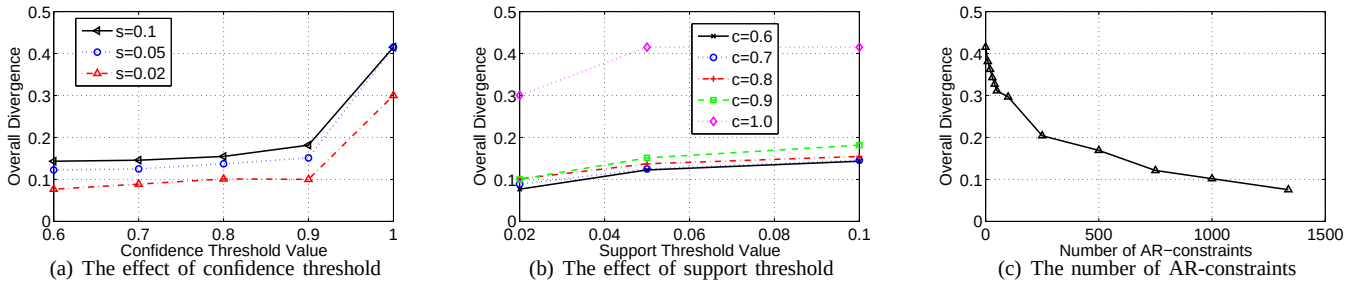
To demonstrate how much sensitive information the association rule mining results disclose, we conduct a series of experiments. We use the Adults dataset from the UC Irvine Machine Learning Repository [31]. We use the following configuration: (1) We remove the records with "?" entries (i.e., incomplete entries). (2) We select the categorical attributes from the dataset, and they are shown in Figure 4. (3) We use the "Salary" attribute as the sensitive attribute. As results, the dataset D has 30162 records, with 7722 distinct QI values and 2 distinct SA values.

	Attribute	**Distinct Values**
1	Workclass	8
2	Marital status	7
3	Occupation	14
4	Relationship	6
5	Race	5
6	Sex	2
7	Native country	41
8	Education	16
9	Salary	2

Fig. 4. UCI Adults

We have implemented our ME method using C++ and Oracle 9i. All experiments were run on an Intel(R) Pentium(R)-D machine with 3.00 GHz CPU and 4GB physical memory. We use the KNITRO software package [24] to solve our Maximum Entropy Estimation problem, which is a special case of Non-Linear Programming problems.

The output of our program is the estimate of $P(SA \mid QI)$ for all the combinations of SA and QI values, given the knowledge of the published association rules. We need to measure how close this estimation is to the distribution in the original dataset. The closer it is to the original distribution, the more private information is disclosed via the published association rule mining results.



(a) The effect of confidence threshold (b) The effect of support threshold (c) The number of AR-constraints

Fig. 5. The impact of association rules

We measure the closeness of the estimated distribution (denoted as $P^*(X \mid Q)$) to the original distribution (denoted as $P(X \mid Q)$). We measure such closeness at two different levels: individual level and overall level. For the individual level, we focus on the estimation for each individual q value, i.e., $P^*(X \mid Q = q)$ for a specific q value. We call $P^*(X \mid Q = q)$ the *estimated individual distribution*. Its closeness to the original distribution $P(X \mid Q = q)$ reveals how much private information of this specific individual is compromised (it could be several individuals who share the same QI value). We use the standard Kullback-Leibler (KL) Divergence [32] to measure the closeness of these two distributions. We call it *Individual Divergence* (denoted by $D_{individual}$):

$$D_{individual} = \sum_{x \in SA} P(x|q) \log \frac{P(x|q)}{P^*(x|q)}.$$

For the overall level, we average the KL-divergence over all possible QI values based on how frequently they appear in the dataset. We call this measure the *Overall Divergence* (denoted by $D_{overall}$):

$$\mathrm{D}_{overall} = \sum_{q \in QI} [P(q) \cdot \sum_{x \in SA} P(x|q) \log \frac{P(x|q)}{P^*(x|q)}].$$

The above two divergence values allow us to understand information disclosure at two different levels. With $D_{individual}$, we can conduct privacy studies for the worst-case scenario, because it allows us to see the result at the individual level; with $D_{overall}$, we can conduct privacy studies for the average-case scenario. As we will show in our experiments, they can tell different things.

A. The impact of association rules

The parameters of threshold s and c play an important role when generating association rules. The smaller they are, the more association rules will be generated. To study how these parameters affect privacy, we measure the overall divergence $D_{overall}$ for a variety of support and confidence values. Figure 5 depicts the results.

In Figure 5(a), we increase the confidence threshold from 0.6 to 1.0. Without any surprise, $D_{overall}$ increases with the increase of confidence threshold, indicating that the overall divergence between the estimated distribution and the original distribution increases. This trend is quite easy to understand: the set of association rules generated with a larger confidence threshold is always a subset of that with a smaller confidence threshold. Therefore, the higher the confidence threshold, the less information will be contained in the published association rules, and the better the privacy can get (i.e., the overall divergence gets larger).

In Figure 5(b), we increase the support threshold from 0.02 to 0.1, and we have observed a similar trend: the overall divergence increases with the increase of the support threshold. The reason of the increase is similar to that of confidence: the higher the support threshold, the fewer association rules will be generated, so, less information will be disclosed, and the privacy will get better.

As we discussed above, the change of the confidence and support thresholds leads to the increase or decrease of the number of association rules. To understand how the number of association rules affect privacy, we plot the overall divergence according to the number of association rules. In this experiment, we fix the support threshold (0.02) and confidence threshold (0.6), and we can get 1337 association rules (of pattern $Q \Rightarrow X$). We sort the association rules by their confidence values in descending order; we then choose the first T association rules, and measure the corresponding $D_{overall}$. We plot T and $D_{overall}$ in Figure 5(c). We clearly see that when the number of AR-constraint increases, the overall divergence value decreases, indicating that the overall privacy gets worse.

B. Publish or withhold exact confidence values

When data owners publish association rules generated from their data, in addition to publishing the confidence and support thresholds, they are tempted to also publish the exact support and confidence values, because they give users more information about those association rules. Obviously, these values contain more information about the original data, and can potentially affect privacy. The question is how severe such an impact is. To answer this question, we have designed an experiment to compare these two situations. We focus on confidence values only; the results for support values are similar.

In the no-release situation, the data owners withhold the exact confidence values; therefore, the AR-constraints generated from the published association rules consist of only

inequalities. The results are plotted as the solid line in Figure 6. In the release situation, the data owners publish the exact confidence values, so the corresponding AR-constraints become equations. The results are plotted as the dotted line in Figure 6. From the figure, we can see a significant difference between these two situations. This interesting result tells us that when the threshold of confidence is small, the decision of whether to publish the exact confidence value has a significant impact on privacy. However, when the confidence threshold gets higher, the impact gets smaller. This trend is quite reasonable, because as the confidence threshold increases, the threshold itself becomes more and more accurate (i.e., the difference between the actual value and threshold becomes narrower); therefore, whether to publish the actual values or not becomes less important.

The results of this experiment give the data owners a useful guideline to decide whether to publish the exact confidence values. In practice, they need to weight the tradeoff between the gain of utility and the loss of privacy. Our method enables them to understand the loss of privacy.

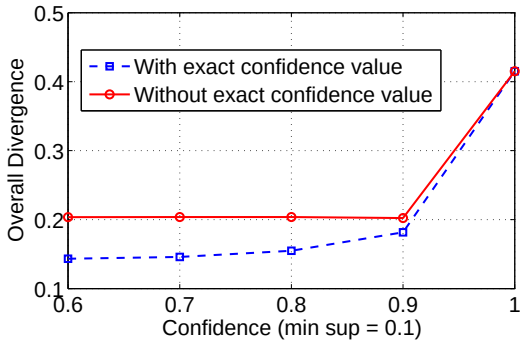


Fig. 6. Publishing or withholding the exact confidence values

C. The impact of NAR-constraints

As we described in Section V, we can derive constraints not only from the published association rules, but also from the non-association rules that are not published because they fail to reach either the confidence or support threshold. We would like to study how much these NAR-constraints affect the privacy of association-rule publishing. To this end, we have conducted two sets of experiments: in one experiment, we do not include the NAR-constraints in our maximum entropy estimation; in the other experiment, we include the NAR-constraints. We plot the difference between these two estimations in Figure 7.

We first study the overall divergence $D_{overall}$ for these two estimations. For each estimation, we calculate the overall divergence between the estimated distribution and the original distribution. We plot the results for both estimations in Figure 7(b). Quite interestingly, we do not see much difference between the two curves, as they almost overlap with each other. This is a surprising result, and it appears that the non-association rules do not carry much information that is detriment to privacy.

To completely understand this surprising result, we decided to look at the *individual divergence* $D_{individual}$, which measures the divergence between the estimated distribution and the original distribution for each individual QI value. We list the Top 10 largest difference of two $D_{individual}$ values in Figure 7(a) (each row correspond to a different QI value). In this dataset, we have two SA values (denoted as SA_1 and SA_2). From the table, we can see the original distribution $P(SA_i \mid QI)$, the estimated distribution with NAR-constraints, and the estimated distribution without NAR-constraints.

For the original distribution, if a QI value q is unique, there is no uncertainty on $P(SA_i \mid Q = q)$. That is why the original probability of $P(SA_i \mid Q = q)$ in the table is either 1 or 0. For these QI values, we can clearly see that with NAR-constraints, the estimated probabilities on the original SA values are more accurate than that without NAR-constraints. For instance, let us take a look at the first row, i.e., the one with the highest difference. We can see that the probability of $P(SA_1 \mid QI)$ without NAR-constraints is 0.762, whereas it is is 0.870 if considering NAR-constraints; this is a 14% increase. This difference is also reflected in the KL-divergence.

To gain a more complete understanding of how NAR-constraints affect the privacy at individual level, we average the top K largest differences between the $D_{individual}$'s obtained with and without NAR-constraints. We let K range from 1 to 500, and the results are plotted in Figure 7(c), which shows that when k becomes large, the average impact of NAR-constraints becomes less significant. From these experiment results, we can say that the impact of NAR-constraints on the overall privacy is not significant, but it can cause a significant difference on certain individuals.

D. Performance Study

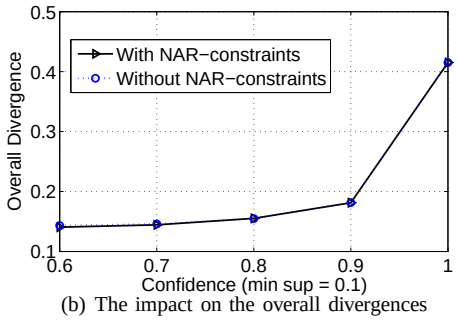
Our method consists of two steps: generating constraints, and conducting Non-Linear Programming. To demonstrate the importance of optimization (i.e. pruning) algorithm, we conduct two different experiments, one using the optimization algorithm and the other without.

We find out that without using the optimization, the computation cannot perform in our machine. The Non-Linear programming solver software runs for about 30 seconds, before it reports an "out of memory" error. The main reason is that the memory is not enough due to the large number of variables (3,900,000) in the constraints.

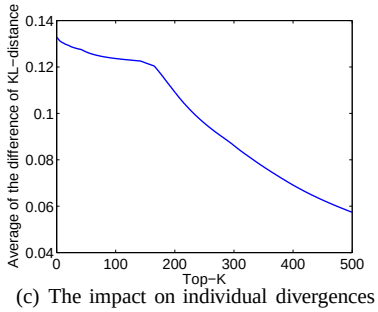
The optimization can decrease the number of variables dramatically. For our dataset, without optimization, we have about 766,000 NAR-constraints (s=0.1, c=0.6); after optimization, there are only 449 NAR-constraints, which have 281,014 variables in total. After the optimization, the solver software can successfully finish. The running time is plotted in Figure 8. The figure is generated with the support threshold 0.1. We choose the confidence thresholds from 0.6 to 0.9. We plot the running time against the number of variables in all constraints, which varies when the confidence changes. In the same figure,

Top-K	**Difference of** KL_1 **and** KL_2	KL_1	KL_2	**Original** $P(SA_1 \mid QI)$	**Original** $P(SA_2 \mid QI)$	$P(SA_1 \mid QI)$ **with NARC**	$P(SA_2 \mid QI)$ **with NARC**	$P(SA_1 \mid QI)$ **without NARC**	$P(SA_2 \mid QI)$ **without NARC**
1	0.133	0.139	0.272	1	0	0.870	0.130	0.762	0.238
2	0.132	0.147	0.279	1	0	0.863	0.137	0.757	0.243
3	0.131	0.155	0.286	1	0	0.856	0.144	0.751	0.249
4	0.131	0.159	0.290	1	0	0.853	0.147	0.748	0.252
5	0.130	0.164	0.294	1	0	0.849	0.151	0.745	0.255
6	0.130	0.164	0.294	1	0	0.848	0.152	0.745	0.255
7	0.130	0.165	0.294	1	0	0.850	0.152	0.745	0.255
8	0.129	0.172	0.301	1	0	0.842	0.158	0.740	0.260
9	0.129	0.165	0.294	1	0	0.848	0.152	0.745	0.255
10	0.129	0.172	0.301	1	0	0.842	0.158	0.740	0.260

(a) The Top-10 largest KL-divergences



(b) The impact on the overall divergences



(c) The impact on individual divergences

Fig. 7. The impact of NAR-constraints on privacy measure

we also plot the running time if only the AR-constraints are used.

From Figure 8, we can see that the computation with NAR-constraints consumes much more time than that without NAR-constraints. Figure 8 also shows that the running time increases when the number of variables increases. But for the case without NAR-constraints, it increases slower than that with NAR-constraints. These observations are consistent with the inherent characteristics of the NLP programming because the search space for inequalities (introduced by the NAR-constraints) is usually larger than that for equalities, and the more constraints and variables we have, the more time it takes to solve the non-linear programming problem.

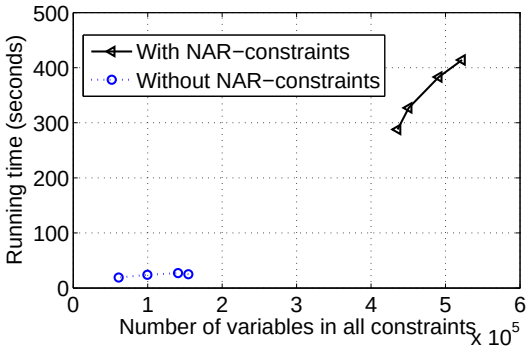


Fig. 8. Running Time

VII. Conclusion and Future Work

We propose a systematic and quantitative analysis for the information disclosure of data mining results. Our method is based on a well-established principle, the Maximum Entropy Principle. We model both association rules and non-association rules as constraints. We then feed these constraints to a non-linear programming software to find the maximum entropy result. To reduce the running time and memory usage, we propose an optimization algorithm to prune all the unnecessary constraints. Our experiment shows that the proposed method is quite effective and efficient.

Several directions of the future work can be followed. One direction is to extend this method to deal with other data mining results, such as decision trees. Another interesting direction is to develop methods to disguise the association rule mining results, such that the privacy requirements are satisfied, while at the same time, the utility of the published results are not compromised too much.

VIII. Acknowledgment

We thank the anonymous reviewers for their comments and kind suggestions.

This work was supported by Awards No. 0312366, 0430252, and 0618680 from the United States National Science Foundation. Any opinions, findings, and conclusions or recommendations expressed here are those of the authors and do not necessarily reflect the views of the funding agencies.

References

[1] R. Agrawal and R. Srikant, "Privacy-preserving data mining," in *Proceedings of the 2000 ACM SIGMOD on Management of Data*, Dallas, TX USA, May 15 - 18 2000, pp. 439–450.

[2] D. Agrawal and C. Aggarwal, "On the design and quantification of privacy preserving data mining algorithms," in *Proccedings of the 20th ACM Symposium on Principles of Database Systems*, Santa Barbara, California, USA, May 21-23 2001.

[3] S. Rizvi and J. R. Haritsa, "Maintaining data privacy in association rule mining," in *Proceedings of the 28th VLDB Conference*, Hong Kong, China, 2002.

[4] A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke, "Privacy preserving mining of association rules," in *Proceedings of 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Edmonton, Alberta, Canada, July 2002.

[5] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan, "Incognito:efficient full-domain k-anonymity," in *Proceedings of the 2005 ACM SIGMOD*, June 12 - June 16 2005.

[6] ——, "Mondrian multidimensional k-anonymity," in *Proceedings of the 22nd IEEE International Conference on Data Engineering (ICDE)*, Atlanta, Georgia, USA, April 2006.

[7] R. J. Bayardo and R. Agrawal, "Data privacy through optimal k-anonymization," in *Proceedings of the 21st IEEE International Conference on Data Engineering (ICDE)*, Tokyo, Japan, April 2005.

[8] B. C. M. Fung, K. Wang, and P. S. Yu, "Top-down specialization for information and privacy preservation," in *Proceedings of the 21st IEEE International Conference on Data Engineering (ICDE)*, Tokyo, Japan, April 2005.

[9] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," Tech. Rep., 1998.

[10] X. Xiao and Y. Tao, "Anatomy: Simple and effective privacy preservation," in *Proceedings of the 32nd Very Large Data Bases conference (VLDB)*, Seoul, Korea, September 12-15 2006, pp. 139–150.

[11] P. Samarati, "Protecting respondents' identities in microdata release," *IEEE transactions on Knowledge and Data Engineering*, no. 6, 2001.

[12] A. Machanavajjhala, J. E. Gehrke, D. Kifer, and M. Venkitasubramaniam, "L-diversity: Privacy beyond k-anonymity," in *Proceedings of the 22nd IEEE International Conference on Data Engineering (ICDE)*, Atlanta, Georgia, USA, April 2006.

[13] N. Li and T. Li, "t-closeness: Privacy beyond k-anonymity and l-diversity," in *Proceedings of the International Conference on Data Engineering (ICDE)*, Istanbul, Turkey, April 17-20 2007.

[14] B.-C. Chen, K. LeFevre, and R. Ramakrishnan, "Privacy skyline: Privacy with multidimensional adversarial knowledge," in *Proceedings of VLDB*, Vienna, Austria, September 23-28 2007.

[15] D. J. Martin, D. Kifer, A. Machanavajjhala, J. E. Gehrke, and J. Halpern, "Worst case background knowledge," in *Proceedings of the 23rd IEEE International Conference on Data Engineering (ICDE)*, Istanbul, Turkey, April 15-20 2007.

[16] X. Xiao and Y. Tao, "m-invariance: Towards privacy preserving republication of dynamic datasets," in *Proceedings of the ACM Conference on Management of Data (SIGMOD)*, 2007.

[17] T. Mielikainen, "On inverse frequet set mining," in *Workshop on Privacy Preserving Date Mining, In conjunction with The Third International Conference on Data Mining 2003*, Melbourne, Florida, November 19 2003.

[18] Z. Wang, W. Wang, and B. Shi, "Blocking inference channels in frequent pattern sharing," in *The 23rd IEEE International Conference on Data Engineering*, Istanbul, Turkey, April 15-20 2007.

[19] Y. Wang and X. Wu, "Approximate inverse frequent itemset mining: Privacy, complexity, and approximation," in *The 5th International Conference on Data Mining 2005*, Houston, Texam, November 27-30 2005.

[20] M. Kantarcioglu, J. Jin, and C. Clifton, "When do data mining results violate privacy?" in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Seattle, Washington, USA, August 22–25 2004.

[21] E. T. Jaynes, "Information theory and statistical mechanics," *Physical Review*, vol. 106, no. 4, pp. 620–630, May 1957.

[22] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, Belmont, Massachusetts, 1995.

[23] A. L. Berger, S. D. Pietra, and V. J. D. Pietra, "A maximum entropy approach to natural language processing," *Computational Linguistics*, vol. 22, no. 1, pp. 39–71, 1996.

[24] R. Byrd, J. Nocedal, and R. Waltz, "Knitro: An integrated package for nonlinear optimization," in *Large-Scale Nonlinear Optimization*, G. di Pillo and M. Roma, Eds. Springer-Verlag, 2006, pp. 35–59.

[25] The TOMLAB Optimization Environment. URL: http://tomopt.com/tomlab/.

[26] W. Du, Z. Teng, and Z. Zhu, "Privacy-MaxEnt: Integrating background knowledge in privacy quantification," in *Proceedings of the ACM SIGMOD International Conference on Management of Data*, Vancouver, BC, Canada, June 9-12 2008, pp. 459–472.

[27] R. Wong, A. Fu, K. Wang, and J. Pei, "Minimality attack in privacy preserving data publishing," in *Proceedings of the 33nd Very Large Data Bases conference (VLDB)*, Vienna, Austria, September 23-28 2007.

[28] L. Zhang, S. Jajodia, and A. Brodsky, "Information disclosure under realistic assumptions: Privacy versus optimality," in *14th ACM Conference on Computer and Communication Security*, Alexandria, Virginia, October 29-November 2 2007.

[29] R. C.-W. Wong, J. Li, A. W.-C. Fu, and K. Wang, "(α, k)-anonymity: An enhanced k-anonymity model for privacy-preserving data publishing," in *Proceedings of ACM KDD*, Philadelphia, Pennsylvania, USA, August 20-23 2006.

[30] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in *Proceedings of International Conference on Very Large Databases (VLDB)*, Santiago,Chile, Sept 1994, pp. 487–499.

[31] UC Irvine Machine Learning Repository. URL: http://www.ics.uci.edu/mlearn/mlrepository.html.

[32] S. Kullback and R. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, pp. 79–86, 1951.