1-1-2008

Privacy-maxent: integrating background knowledge in privacy quantification

Wenliang Du
Syracuse University, Department of Electrical Engineering and Computer Science, wedu@syr.edu

Zhouxuan Teng
Syracuse University, Department of Electrical Engineering and Computer Science, zhteng@syr.edu

Zutao Zhu
Syracuse University, Department of Electrical Engineering and Computer Science, zuzhu@syr.edu

Privacy-MaxEnt: Integrating Background Knowledge in Privacy Quantification

Wenliang Du, Zhouxuan Teng, and Zutao Zhu
Department of Electrical Engineering and Computer Science
Syracuse University, Syracuse, NY 13244
Tel: 315-443-9180 Fax: 315-443-1122
{wedu,zhteng,zuzhu}@syr.edu

ABSTRACT

Privacy-Preserving Data Publishing (PPDP) deals with the publication of microdata while preserving people' private information in the data. To measure how much private information can be preserved, privacy metrics is needed. An essential element for privacy metrics is the measure of how much adversaries can know about an individual's sensitive attributes (SA) if they know the individual's quasi-identifiers (QI), i.e., we need to measure $P(SA \mid QI)$. Such a measure is hard to derive when adversaries' background knowledge has to be considered.

We propose a systematic approach, Privacy-MaxEnt, to integrate background knowledge in privacy quantification. Our approach is based on the maximum entropy principle. We treat all the conditional probabilities $P(SA \mid QI)$ as unknown variables; we treat the background knowledge as the constraints of these variables; in addition, we also formulate constraints from the published data. Our goal becomes finding a solution to those variables (the probabilities) that satisfy all these constraints. Although many solutions may exist, the most unbiased estimate of $P(SA \mid QI)$ is the one that achieves the maximum entropy.

Categories and Subject Descriptors

H.2.7 [**Database Management**]: Database Administration—*Security, integrity, and protection*

General Terms

Security

Keywords

Privacy quantification, data publishing

1. INTRODUCTION

Privacy-Preserving Data Publishing (PPDP) achieves microdata publishing while preserving individuals' private information. In PPDP, the original data D usually consists of three types of attributes: *identifiers (ID)*, *quasi-identifiers (QI)*, and *sensitive attributes (SA)*. The ID attributes consist of people's identity information, such as names and social security numbers. This information will definitely be removed from the published data for privacy reasons. The QI attributes usually include people's demography information, such as gender, zip code, age, etc. QI is not considered as sensitive information, because it can also be obtained from other sources [22]. The SA attributes include sensitive information about individuals. For example, in a medical data set, patients' diseases and diagnoses belong to SA.

Although SA contains sensitive information, it does not disclose any individual's private information if it cannot be linked to an individual. However, publishing QI along with SA (without any disguising) can make the linking possible. This is because the QI can help adversaries re-identify individuals, even though the actual identifiers are removed [22]; after re-identification, linkings between SA and individuals can be established. This is the primary risk faced by data publishing. This type of attack is called *linking attacks*. The objective of PPDP is to publish a transformed microdata D' in a way that minimizes the risk of linking attacks, while maximizing the usefulness of the original data D.

A number of PPDP methods have been proposed, including generalization [13, 14, 4, 10, 22], randomization [3, 1, 21, 9], and bucketization [25, 19]. In this paper, we focus on the bucketization method.

The Bucketization method is proposed by Xiao and Tao [25] and further studied by Martin et al [19]. In this method, the records of the dataset are partitioned into *buckets*. Within each bucket, the SA attributes of all the records are mixed together to break the bindings between QI and SA; therefore each QI can be potentially binded to multiple SA values. Figures 1(a) and 1(b) list an example of original data set and its bucketization result. We will use these sample data sets throughout this paper.

Privacy Quantification and Background Knowledge. To understand how much privacy is preserved in data publishing, we need to quantify privacy. Since linking attacks are the primary risks in PPDP, we need to quantify how much information adversaries can know in linking attacks. This is essentially to derive the conditional probability $P(SA \mid QI)$, for any instance of SA and QI. This probability is essential for various privacy quantification metrics, such as L-diversity [17].

Most of the existing metrics assumes that adversaries do not have any background knowledge. In reality, this might not be a valid assumption. For example, common medical

SIGMOD'08, June 9–12, 2008, Vancouver, BC, Canada.

Name	Gender	Degree	Disease
Allen	male	college	Flu
Brian	male	college	Pneumonia
Cathy	female	college	Breast Cancer
David	male	high school	Flu
Ethan	male	college	HIV
Frank	male	high school	Pneumonia
Grace	female	junior	Breast Cancer
Helen	female	college	HIV
Iris	female	graduate	Lung Cancer
James	male	graduate	Flu

(a) The original data set D

Gender	Degree	Disease	Bucket
male	college	{Breast cancer, Flu Flu, Pneumonia}	1
male	college		
female	college		
male	high school		
male	college	{Breast cancer, HIV Pneumonia}	2
male	high school		
female	junior		
female	college	{Flu, HIV, Lung cancer}	3
female	graduate		
male	graduate		

(b) The bucketized data set D'

Quasi-Identifier	Sensitive Attribute	Bucket
q_1	$\{s_1, s_2, s_2, s_3\}$	1
q_1		
q_2		
q_3		
q_1	$\{s_1, s_3, s_4\}$	2
q_3		
q_4		
q_2	$\{s_2, s_4, s_5\}$	3
q_5		
q_6		

(c) D' in abstract form

Figure 1: The data example used throughout this paper

knowledge tells us that it is rare for `male` to have `Breast Cancer`. With this knowledge, for the bucketized data D' listed in Figure 1(b), we immediately know that both females in Bucket 1 and Bucket 2 have `Breast Cancer`, because they are the only females in their respective buckets.

Background knowledge can come with different forms, for example, it might be a rule like the previous `Breast Cancer` example; it might be a probability, such as $P(s \mid q) = 0.2$ for certain $s \in SA$ and $q \in QI$; it might even be an inequality, such as $0.3 \leq P(s \mid q) \leq 0.5$. To make things even more complicated, it can be about individuals. For example, adversaries might know that "Frank has `Pneumonia`", or "either Iris or Brian has `Lung Cancer`".

With these kinds of background knowledge, computing $P(SA \mid QI)$ is extremely difficult. Only a few studies have been conducted in integrating background knowledge in privacy quantification [19, 7]. However, the existing work is not generic enough to deal with the large variety of background knowledge.

Our Approach. We propose a generic and systematic method to integrate background knowledge in privacy quantification, which can deal with many different types of background knowledge described above. We call our method *Privacy-MaxEnt*. In this method, we formulate the derivation of $P(SA \mid QI)$ as a non-linear programming problem. We treat $P(S \mid Q)$ as a variable for each combination of $S \in SA$ and $Q \in QI$. Our goal is to assign probability values to these variables. We treat background knowledge as constraints, i.e., the assignment of the variables must be consistent with the background knowledge. We also formulate the published data set D' as constraints for those variables. Therefore, the derivation of $P(S \mid Q)$ becomes finding an assignment for these variables that satisfy the constraints.

Because the number of variables in PPDP often outnumbers the number of constraints, many solutions are possible. However, since the meaning of each variable is actually our inference on $P(S \mid Q)$, we want such an inference as unbiased as possible. The principle of Maximum Entropy states that *when the entropy of these variables is maximized, the inference is the most unbiased*. Therefore, applying the Maximum Entropy principle, our problem becomes finding the maximum-entropy assignment for those variables that satisfy the constraints. *MaxEnt* in the name of our scheme, *Privacy-MaxEnt*, stands for Maximum Entropy.

We face two challenges in applying the Maximum Entropy principle to solve the privacy quantification problem. First, we need to be able to formulate the background knowledge as linear constraints; such formulation should be generic enough to accommodate various types of background knowledge. Second, being another source of constraints, the published data set D' also needs to be formulated as linear constraints. For this type of constraints, we have to ensure that not only the constraints are correct (i.e, sound), they also need to be complete, i.e., all the knowledge from D' has to be formulated as constraints.

There is a third challenge that we have to face. It is not specific to our approach; it is a challenge that all the work on integrating background knowledge has to face. That is, when quantifying privacy, how to determine what and how much background knowledge that adversaries might have. This is a difficult task, because we cannot predict what adversaries might know. Instead of predicting, we propose a mechanism to specify the bound of background knowledge, and quantify privacy under the assumption of the bound. Therefore, the outcome of privacy quantification should be a tuple consisting of bound and privacy score. It is up to the users (those who want to publish their data) to decide what bound is acceptable to them. Our goal is to provide a mechanism for users to specify their assumptions (i.e. acceptable bounds), and then tell them how much private information will be preserved in their published data, when adversaries' amount of background knowledge is within those bounds.

Organization. In Section 2, we discuss the work related to this paper. In Section 3, we give a brief introduction of the Maximum Entropy Principle, and how we formulate our problem as an Maximum Entropy modeling problem. In Section 4, we present how background knowledge can be modeled as linear constraints, and how to specify the bound of background knowledge. In Section 5, we formulate linear constraints from the published data set, and prove that the constraints are sound and complete. In Section 6, we describe how our approach can be extended to model a variety of knowledge types. In Section 7, we conduct experiments to evaluate our approach. Finally, in Section 8, we conclude and discuss future work.

2. RELATED WORK

In the early studies of the privacy in privacy-preserving data publishing, background knowledge is not considered. A number of privacy quantification methods have been proposed, including K-anonymity [22], L-diversity [17], (α, k)-anonymity [24], t-Closeness [15], etc.

Martin et al. [19] proposed the first formal study of the effect of background knowledge on privacy-preserving data publishing. In this work, background knowledge is formulated as conjunctions of k basic implicators. Each basic implication is a rule specifying the implication relationship between two atoms, and each atom is a predicate about a person and his/her sensitive values. The authors then use k to bound the background knowledge, and compute the maximum disclosure of a bucket data set with respect to the background knowledge.

The work in [19] is further improved by Chen et al. in a scheme called privacy skyline [7]. Realizing the limitation of using a single number k to bound background knowledge, the authors in [7] use a triple (ℓ, k, m) to specify the bound of the background knowledge about a particular person (called target). Namely, the bound specifies that (1) adversaries know ℓ other people's sensitive value; (2) adversaries know k sensitive values that the target does not have; (3) adversaries know a group of $m-1$ people who share the same sensitive value with the target. Based on this triple, the authors propose a method to incorporate background knowledge in privacy quantification.

Both works in [19] and [7] are significant in the sense they pioneer the treatment of background knowledge in PPDP. However the major shortcomings of these two papers is the lack of power in expressing background knowledge. The language used in these papers can express the background knowledge using deterministic rules; as the authors in [19] point out, probabilistic background knowledge cannot be expressed using the current language. Even if the language can be extended to probabilistic knowledge, the algorithms to quantify privacy will have to be modified significantly.

Our work specifically targets the probabilistic background knowledge. Our language is much more generic than the existing work; as long as background knowledge can be expressed as linear equations or linear inequalities of probabilities, it can be integrated in privacy quantification. Moreover, our proposed method is systematic; namely, regardless of how complicated those linear equations and inequalities are and how many they are, they are just the inputs; the algorithm used to quantify privacy is always the same.

3. MAXIMUM ENTROPY MODELING

3.1 The Privacy Quantification Problem

To facilitate the explanation, we use the data set depicted in Figure 1(c) throughout this paper. This is the same bucketized data set as the one depicted in Figure 1(b), but in an abstract form (to simplify presentation). In this data set, each q_i represents a unique instance (or value) of the QI attributes, and each s_i represents a unique instance of sensitive attributes. If two people have the same QI value, their QI values will be denoted by the same symbol. For example, q_1 represents `{male, college}`, and it appears three times in the data.

To be able to quantify the privacy of a bucketized data when certain background knowledge is present, it is important to calculate $P(S \mid Q)$ for all the combinations of Q and S, where Q represents the QI attributes, and S represents the SA attributes. Assume that there are m buckets, and let B represent the bucket index. We can use the following formula to calculate $P(S \mid Q)$:

$$\begin{aligned} P(S \mid Q) &= \frac{P(Q,S)}{P(Q)} = \frac{1}{P(Q)} \cdot \sum_{B=1}^{m} P(Q,S,B) \\ &= \frac{1}{P(Q)} \cdot \sum_{B=1}^{m} P(S \mid Q,B) \cdot P(Q,B), \end{aligned}$$

where $P(Q)$ is the distribution of the QI attributes; since in bucketized data, the QI attributes are not disguised, we can directly get $P(Q)$ and $P(Q,B)$ from the bucketized data set. Therefore, $P(S \mid Q,B)$ is the only thing that needs to be derived. Without background knowledge, $P(S \mid Q,B)$ can be computed quite easily:

$$P(S \mid Q,B) = \textit{Portion of } S \textit{ in bucket } B. \quad (1)$$

The key assumption made in the above equation is the following: given Q in a bucket B, the probability that this Q corresponds to a sensitive value is *uniform* across all the possible sensitive values within the bucket B. This is a reasonable assumption if D' is the only available knowledge about D. However, when other types of knowledge are available, this assumption can become invalid. For example, in the example from Figure 1(c), if the adversaries know that $P(s_1 \mid q_2) = 0$ and $P(s_1 \text{ or } s_2 \mid q_3) = 0$, we immediately know that in the first bucket, q_3 can only be mapped to s_3, q_2 can only be mapped to s_2, and one of the q_1 maps to s_1 and the other maps to s_2.

Directly computing $P(S \mid Q,B)$ can become quite difficult when the background knowledge becomes complicated. For example, if $P(s_1 \mid q_3) = 0.2$, instead of 0, computing $P(S \mid Q, B=1)$ has to consider all the buckets that contain both q_3 and s_1. Similarly, if we know that $P(s_1 \mid q_3) + P(s_2 \mid q_3) = 0.4$, we have to consider all the buckets that contain s_1, s_2, and q_3.

To summarize the above discussions, the fundamental question that we are trying to solve is to assign values to $P(S \mid Q,B)$ for each combination of Q, S, and B, such that they are consistent with the background knowledge and the information contained in the bucketized data.

3.2 Maximum Entropy Modeling

It is possible that many assignments of $P(S \mid Q,B)$ are consistent with the given knowledge and data, with some being biased toward certain particular S values. Being biased means assuming some extra information that we do not posses; therefore, the least biased assignment is the most desirable. It is widely believed that the least biased distribution that encodes certain given information is that which maximizes the information entropy [6]. This is the principle of the *Maximum Entropy (ME)*, which was first expounded by Jaynes in 1957.

Applying the ME principle, our problem becomes computing $P(S \mid Q,B)$ for all possible values of Q, S, and B, such that the following conditional entropy $H(S \mid Q,B)$ is maximized:

$$\begin{aligned} & H(S \mid Q,B) \\ = & -\sum_{Q,S,B} P(Q,B)P(S \mid Q,B) \log P(S \mid Q,B). \end{aligned} \quad (2)$$

Because $H(S \mid Q,B) = H(Q,S,B) - H(Q,B)$ and also because $H(Q,B)$ is a constant given the published data, maxi-

mizing $H(S \mid Q, B)$ is equivalent to maximizing $H(Q, S, B)$:

$$H(Q, S, B) = -\sum_{Q,S,B} P(Q, S, B) \log P(Q, S, B). \quad (3)$$

Namely, our goal becomes computing $P(Q, S, B)$ for all possible values of Q, B, and S, such that $H(Q, S, B)$ is maximized. [1]

Without any constraint, $H(Q, S, B)$ is maximized when $P(Q, S, B)$ has a uniform distribution. However, we do have two different sources of constraints. One is the published data D', i.e, the distribution of $P(Q, S, B)$ should be consistent with the published data D'. For example, if in D', bucket i does not contain s, the value of $P(Q, s, i)$ must be zero for all Q's in bucket i. The other source of constraints is the background knowledge, i.e., the distribution of $P(Q, S, B)$ should also be consistent with the background knowledge. For instance, if the background knowledge tells us that the binding of q and s is impossible, $P(q, s, B)$ should be zero for all buckets.

To apply the ME principle, we need to formulate all the constraints as linear equations based on $P(Q, S, B)$. Let these constraints be h_1, ..., h_w. Our problem can be formally defined as the following:

DEFINITION 3.1. *(Maximum Entropy Modeling) Finding an assignment for $P(Q, S, B)$ for each combination of Q, S, and B, such that the entropy $H(Q, S, B)$ is maximized, while all the linear equations h_1, ..., h_w are satisfied.*

3.3 Solving the ME Problem

The maximum entropy problem is a non-linear programming problem with equality constraints. It is a special case of the following general form:

$$\text{minimize } f(\vec{x}),$$
$$\text{subject to } h_i(\vec{x}) = 0, \quad i = 1, \ldots, w,$$

where $f : \Re^n \mapsto \Re$ and $h_i : \Re^n \mapsto \Re$, $i = 1, \ldots, w$, are continuously differentiable functions. Function f is called the objective function, while functions h_i's are called the constraint functions.

In the Maximum Entropy problem, the $P(Q, S, B)$ for each instance of (Q, S, B) is considered as one dimension of the variable x. For example, if we have 1000 different combinations of (Q, S, B), x will be a vector of 1000 dimensions. Similarly, each function h_i, which is a linear function of x, is actually a linear function of various $P(Q, S, B)$ instances.

There are existing methods to solve this kind of optimization problem. We refer readers to [5, 6] for details. The most common solution is to apply the method of *Lagrange multipliers* to convert this *constrained* optimization problem to an *unconstrained* optimization problem, which can then be solved using various numerical methods, such as steepest ascent, Newton's method, and LBFGS (limited-memory quasi-Newton packages for large scale optimization) [16]. There are also several numerical methods specifically tailored to the maximum entropy problem, including the generalized [8] and the improved [20] iterative scaling algorithms. A comparison of various methods for solving the maximum entropy problem is given by Malouf [18].

[1] We compute $P(Q, S, B)$, instead of $P(S \mid Q, B)$, because it is easier to formulate constraints using $P(Q, S, B)$. Deriving $P(S \mid Q, B)$ from $P(Q, S, B)$ is trivial because $P(Q, B)$ can be directly obtained from the published data.

3.4 Sources of ME Constraints

To apply the ME principle to solve our problem, we need to convert all the available knowledge into constraints. These constraints, due to the intrinsic property of maximum entropy modeling, must be in the form of linear equations. In PPDP, constraints come from two different sources. One is the published data set itself; namely, although disguised, the published data still reveal some information about the original data. We need to abstract that information from the published data, and formulate it as constraints. The other source of constrains include everything other than the published data set; we call this source the background knowledge. We describe how to derive constraints from these two different sources in the next three sections.

4. BACKGROUND KNOWLEDGE

As we mentioned in the previous section, as long as we can represent background knowledge using linear equations based on $P(Q, S, B)$, we can use the ME method to integrate background knowledge in the calculation of $P(S \mid Q)$, which can then be further used to quantify privacy. We call these linear equations ME constraints.

Background knowledge can have many different forms. They can be classified into two categories: knowledge about data and knowledge about individuals. Knowledge about data consists of knowledge about the data distribution. An example of this type of knowledge is the following: "it is rare for male to have breast cancer". Knowledge about individuals consists of knowledge about individual people. Here are a few examples: "Bob does not have HIV", "one of Alice or Bob must have Cancer", "Charlie either has Cancer or HIV". This paper mainly focuses on knowledge about data distributions. However, in Section 6, we extend our work to the knowledge about individuals.

4.1 Knowledge About Data Distribution

Most of this type of knowledge can be expressed using conditional probability $P(S \mid Q_v)$, where Q_v is a subset of the QI attributes. The previous example about breast cancer can be written as P(`Breast Cancer` | `Male`) = 0. Because ME constraints only contain joint distribution of QI attributes (Q), SA attributes (S), and Bucket index (B), we need to convert $P(S \mid Q_v)$ to $P(Q, S, B)$. We first convert it to $P(Q_v, S)$ as the following:

$$P(Q_v, S) = P(S \mid Q_v) \cdot P(Q_v),$$

where $P(Q_v)$ is the probability of people who have Q_v. This knowledge is about the people in the entire population. It is difficult to get $P(Q_v)$, but we can use the sample distribution in the published data set to approximate this distribution. Therefore, in this paper, we use $P(Q_v)$ to represent the probability of Q_v in the sample data set.

We do need the entire QI attributes in our ME constraints, not a subset of it. For example, if the data set has 3 quasi-identifier attributes, `Gender`, `Degree`, and `Age`, then all these three QI attributes must appear in our ME constraints, in particular, in the probability expressions used in those constraints. For instance, we cannot have a constraint like the following: $P(Flu \mid \texttt{Gender} = male) = 0.3$. We have to convert such a probability expression into one that includes all three attributes.

Let Q represent the set of entire QI attributes, and let $Q_- = Q - Q_v$ be the difference between Q and Q_v. Therefore

$P(Q, S)$ can be written as $P(Q_v, Q_-, S)$. By summing up the this probability over all possible Q_- values, we have the following:

$$\sum_{Q_-} P(Q_v, Q_-, S) = P(Q_v, S).$$

We also need to include the bucket index B in ME constraints, although background knowledge does not depend on any bucketization schemes. Namely, the constraints should be the same regardless how the published data are bucketized. We have the following:

$$\sum_{B=1}^{m} \sum_{Q_-} P(Q_v, Q_-, S, B) = P(Q_v, S) = P(S \mid Q_v) \cdot P(Q_v).$$

This is the final ME constraint for the background knowledge of $P(S \mid Q_v)$. As long as any background knowledge can be presented as conditional probability, it can be formulated as an ME constraint.

Let us see an example. Assume that the knowledge says $P(Flu \mid male) = 0.3$ (this is fictitious). Based on the data set in Figure 1, we can construct the following constraint for this knowledge: $P(\{male, college\}, Flu, 1) + P(\{male, highschool\}, Flu, 1) + P(\{male, college\}, Flu, 3) + P(\{male, graduate\}, Flu, 3) = 0.3 * P(male) = 0.3 * 6/10 = 0.18$.

4.2 Where to get Background Knowledge

An inevitable problem we face is where to get background knowledge. The common knowledge about male and breast cancer comes from our knowledge about medical field. Are we supposed to know every common knowledge in order to represent background knowledge? The answer is no.

The best source of knowledge about data distribution is actually contained in the original data D itself; any knowledge that is inconsistent with D is incorrect, regardless of whether it is true in general or not. For example, although it is rare for male to get breast cancer, it is still possible, and the data D might contain the records of such a rare case. Therefore, in this case, although the common knowledge is true in general, it is actually incorrect for this specific data set; adversaries can be misled by such incorrect knowledge.

Therefore, there is no need for us to learn all the knowledge about the world, people, medicine, health, finance, etc. All we need is to derive the background knowledge from the original data. After all, the goal of adversaries is to derive the links between QI and SA in the original data from a disguised published data. The more background knowledge they know about the original data, the more accurately they can derive the relationship between QI and SA.

4.3 What Background Knowledge to Use

Finding what background knowledge to use in privacy quantification is a very challenging problem, because it requires us to predict what and how much background knowledge the adversaries might know. This is an infeasible task. Let us take a step back, and think about why we analyze the impact of background knowledge while having no idea what adversaries might know. The reason is that we want to understand the privacy of a disguised data set under various assumptions. For the most of the existing privacy quantification schemes, the implicit assumption made by them is that the adversaries have no background knowledge at all.

Understanding the assumption is important for users to understand the privacy of their published data. Therefore, the outcome of privacy quantification should be a tuple consisting of the assumptions about background knowledge and the privacy score. Users can understand the risk of their data publishing under various assumptions. They will be able to judge whether the assumptions are too strong or not; if not, whether the privacy score under the assumptions are acceptable. Therefore, our task is not to predict what adversaries might know; instead, we should be presenting to the users a more complete understanding of the privacy of their data that they plan to publish.

The challenge becomes finding a way to specify assumptions about background knowledge. One approach is to enumerate the knowledge as sets of different size. There will be many assumptions, because of the combinations of knowledge are too many. Another approach is to present a bound of knowledge. The bound specifies the amount of background knowledge that we assume that adversaries can have. Therefore, our privacy quantification is tied with the bound. Various bound can be used. In this paper, we propose a bound called Top-(K_+, K_-) strongest associations.

4.4 Top-(K_+, K_-) Strongest Associations

As we described earlier, the knowledge about data distribution can be modeled as a set of conditional probabilities between QI attributes and SA attributes, i.e., $P(S \mid Q)$, for $S \in SA$ and $Q \in QI$. However, not all conditional probabilities can be considered as knowledge. A conditional probability $P(S \mid Q)$ becomes knowledge if this probability is sufficiently high. This is similar to the association rule concept. According to association rule mining [2], if $P(S \mid Q)$ is sufficiently high (called *confidence*), and $P(Q, S)$ is also large enough (called *support*), we say that $Q \Rightarrow S$ is an association rule. Therefore, we can use the association rules between Q and S as our knowledge.

The above association rules are called *positive association rules*; another type of association rules has the form of $Q \Rightarrow \neg S$, meaning that when a person has Q as its Quasi-Identifier, he/she is unlikely to have S. The `Breast Cancer` example we use is actually this type of association. This is called *negative association rule* [23]. Several other types of rules are also called negative association rule, such as $\neg Q \Rightarrow S$ and $\neg Q \Rightarrow \neg S$.

We use the number of association rules as the bound of background knowledge. We call this bound the Top-(K_+, K_-), where K_+ represents the top K_+ positive association rules and K_- represents the top K_- negative association rules. Namely, we derive all the possible positive and negative associations between Q and S. For each type of association, we sort them based on their confidence levels; we then pick the top K_+ positive association rules and the top K_- negative association rules. We use these two sets of association rules as the background knowledge. These two sets include the most useful background knowledge about the data distribution (not individuals); if available for linking attacks, they can be quite helpful.

4.5 Inequality Background Knowledge

Equation allows us to express accurate background knowledge; in reality, background knowledge can be vague, and equations are not good for expressing vagueness. For instance, equations cannot express the fact that $P(s_1 \mid q_1)$

Bucketized Data	
QI	**SA**
q_1	$\{s_1, s_2, s_2, s_3\}$
q_1	
q_2	
q_3	

Assignment 1	
QI	**SA**
q_1	s_1
q_1	s_2
q_2	s_2
q_3	s_3

Assignment 2	
QI	**SA**
q_1	s_3
q_1	s_2
q_2	s_1
q_3	s_2

Figure 2: Assignments: bucketized data and different ways to assign (or associate) SAs to QIs.

is about (not exactly) 0.3. Inequalities can help expressing vague knowledge. In the previous example, we can state $0.3 - \epsilon \leq P(s_1 \mid q_1) \leq 0.3 + \epsilon$, where ϵ represents the degree of vagueness. In addition to expressing vague knowledge, inequalities can also express inequality relationship. For example, the background knowledge might state that a person with q_1 attribute is more likely to have s_1 diseases than to have s_2 diseases. We can express this knowledge as $P(s_2 \mid q_1) < P(s_1 \mid q_1)$.

Kazama and Tsujii has extended the Maximum Entropy Modeling to deal with inequality constraints [11]. Using this extension, in addition to equation constraints, we can now include the background knowledge that can only be modeled as inequalities. This makes the ME model more powerful in modeling background knowledge. Moreover, using inequalities, we can add the degree of vagueness (ϵ) to the bound of background knowledge (equations assume $\epsilon = 0$). We will further study the extended ME model in our future work. In this paper, we only focus on equality constraints.

5. CONSTRAINTS FROM THE DATA

5.1 Definitions

DEFINITION 5.1. *(Probability Term and Expression) Assume that there are m buckets in a bucketized data set D'. A probability $P(q, s, b)$ is called a* probability term *if q is an instance of the QI attributes, s is an instance of the SA attribute, and b is a bucket index (i.e. $b \in [1, m]$). We call the linear combination of probability terms the probability expression.*

For example, for D' in Figure 1(c), $P(q_1, s_1, 1)$ is a probability term, and $P(q_1, s_1, 1) + P(q_1, s_2, 1) + P(q_1, s_3, 1)$ is a probability expression.

DEFINITION 5.2. *(Assignment for a bucket) Assignment for the bucket b is a set of tuples: $\Lambda(b) = \{(q, s) \mid q \in QI(b), s \in SA(b)\}$, where each instance of q and s appears once and only once in $\Lambda(b)$. It should be noted that certain QI values (or SA values) might appear multiple times in a bucket; they are treated as multiple instances.*

Figure 2 gives two different assignments for the bucketized data list in the leftmost table (this bucketized data set is taken from the Bucket 1 of Figure 1(c), and we omit the bucket index column). In the first assignment, $\Lambda(b = 1) = \{(q_1, s_1), (q_1, s_2), (q_2, s_2), (q_3, s_3)\}$. In the second assignment, $\Lambda(b = 1) = \{(q_1, s_3), (q_1, s_2), (q_2, s_1), (q_3, s_2)\}$. Because both q_1 and s_2 appear twice in the bucket, they also appear twice in the assignment.

DEFINITION 5.3. *(Assignment for the bucketized data) Assignment (Λ) for the entire bucketized data is the following:*

$$\Lambda = \bigcup_{b=1}^{m} \Lambda(b).$$

For a probability expression F, we use $F(\Lambda)$ to represent the value of F under the assignment Λ.

DEFINITION 5.4. *(Invariant) A probability expression F is an invariant if $F(\Lambda)$ is a constant for any Λ of D'.*

We use examples to explain the above definition. Assume that the entire disguised data set D' used in Figure 2 only contains one bucket, i.e., the total number of records is 4. It is not difficult to see that $F = P(q_1, s_1, 1)$ is not an invariant: it equals 0.25 in the first assignment, and 0 in the second assignment. On the other hand, we know that $F = P(q_1, s_1, 1) + P(q_2, s_1, 1) + P(q_3, s_1, 1)$ is an invariant, because it does not matter how QI and SA are assigned to each other, F is always equal to the portion of s_1 among all the SA values in the Bucket 1, which is 0.25 (1 out of 4). Based on invariants, we can define invariant equations.

DEFINITION 5.5. *(Invariant Equation) An invariant equation has the form of $F = C$, where F is an invariant, C is a constant, and $F = C$ always holds true for all the assignments for D'. We also call invariant equations* ME constraints*, because they are used as constraints by ME.*

Since the invariant equations are always true among all the assignments, and the original data D can be considered as one of these assignments, invariant equations are the only things that we can say about the original data D with absolute certainty; everything else is an inference. For example, past work on privacy analysis often says that $P(q_1, s_1, 1) = 0.25$. This is only an inference; if there is no background knowledge, this inference is unbiased, and is acceptable to privacy analysis. However, with background knowledge, inferences like this may become biased and unacceptable.

To apply ME, we can only formulate invariant equations, not inferences, as constraints. The objective of the maximum entropy estimate is to derive the maximum entropy inferences that satisfy all the constraints; using an inference as a constraint requires ME to preserve this pre-determined inference during the computation. Unless such an inference already achieves the maximum entropy, preserving it can fail to achieve the maximum entropy.

Furthermore, to apply ME, not only do we need to identify the invariant equations from the disguised data D', we need to identify all of them. These equations are the knowledge that we need in ME; if one invariant equation is missing, we are not using all the knowledge from D'. Formally speaking, the set of invariant equations must be *complete*.

In the rest of this section, we will derive invariant equations from D', and then we will prove several interesting properties of these invariants, including soundness, conciseness, consistency, and most importantly, *completeness*.

5.2 Finding Invariants

We will only focus on finding *within-bucket invariants*. A within-bucket invariant consist of the probability terms from the same bucket. Using the following lemma, we will

show that any invariant can be written as the sum of within-bucket invariants. Therefore, it is sufficient to just identify the within-bucket invariants.

LEMMA 1. *Let F represent a linear combination of probability terms of the published data D'. We write F as $F_1 + ... + F_m$, where F_i contains the probability terms only from bucket i. F is an invariant for D', if and only if $F_1, \ldots, F_m$ are also invariants for D'.*

PROOF. We only prove that if F is an invariant for D', then each F_i is also an invariant; the other direction is trivial. Assume at least one of F_i's is not an invariant, we will prove that F is not an invariant either. Without the loss of generality, we assume that F_1 is not an invariant; therefore, there exists two different assignments that cause the values of F_1 to be different. Let $\Lambda(1)$ and $\Lambda'(1)$ be these two assignments in the Bucket 1, so we have $F_1(\Lambda(1)) \neq F_1(\Lambda'(1))$. We then construct the following two assignments:

$$\Lambda_a = \{\Lambda(1), \Lambda(2), \ldots, \Lambda(m)\}$$
$$\Lambda_b = \{\Lambda'(1), \Lambda(2), \ldots, \Lambda(m)\},$$

where the only difference between Λ_a and Λ_b is the assignments in Bucket 1, i.e., $\Lambda(1) \neq \Lambda'(1)$.

Since $\Lambda(1)$ and $\Lambda'(1)$ do not affect the probability terms in the other buckets, we have $F_i(\Lambda_a) = F_i(\Lambda_b)$ for $i = 2, \ldots, m$. At the same time, we know that $F_1(\Lambda_a) \neq F_1(\Lambda_b)$. Since $F(\Lambda)$ is the sum of $F_i(\Lambda)$, for $i = 1, \ldots, m$, we can conclude that $F(\Lambda_a) \neq F(\Lambda_b)$; in other words, F is not an invariant. This is contradictory to the fact in the lemma. Therefore, $F_1, \ldots, F_m$ all need to be invariants. □

Because of the above lemma, we will only focus on the within-bucket invariants in the rest of this paper. Suppose there are m buckets in total, and we focus on the Bucket b, where $b = 1, \ldots, m$. We use $QI(b) = \{q_1, \cdots, q_g\}$ to represent the set of *distinct* QI values in Bucket b, where g is the size of $QI(b)$. Similarly, we use $SA(b) = \{s_1, \cdots, s_h\}$ to represent the set of *distinct* SA values in Bucket b, where h is the size of $SA(b)$. We have identified three types of invariant equations: *QI-invariant*, *SA-invariant*, and *Zero-invariant* equations.

QI-invariant Equation. The following equation is called *QI-invariant* equation. There are g such equations for bucket b (one for each $q \in QI(b)$):

$$\sum_{j=1}^{h} P(q, s_j, b) = P(q, b), \text{ for } q \in QI(b). \quad (4)$$

The probability $P(q, b)$ is the joint probability of QI value q and bucket index value b in the dataset D'. Since D' is given, this value is a constant, and can be directly obtained from D'. Therefore, the right side of the above equation is a constant, and the left side of the equation is called QI-invariant. An example of QI-invariant for the data in Figure 1(c) is the following:

$$P(q_1, s_1, 1) + P(q_1, s_2, 1) + P(q_1, s_3, 1) = P(q_1, 1) = \frac{2}{10}.$$

SA-invariant Equation The following equation is called *SA-invariant* equation. There are h such equations for bucket b (one for each $s \in SA(b)$):

$$\sum_{i=1}^{g} P(q_i, s, b) = P(s, b), \text{ for } s \in SA(b). \quad (5)$$

Similar to the QI-invariant, the probability $P(s, b)$ at the right side of the equation is also a constant, and can be directly obtained from D'. The left side of the equation is called SA-invariant. Both the QI-invariant and SA-invariant equations capture the fact that although we cannot tell how each QI's value match with SA's value in a bucket for different assignments, we are sure that the total probability of matching for one value of the QI or SA is fixed. These values can be directly counted from the disguised data set. Here is an example of SA-invariant for the data in Figure 1(c):

$$P(q_1, s_4, 2) + P(q_3, s_4, 2) + P(q_4, s_4, 2) = P(s_4, 2) = \frac{1}{10}.$$

Zero-invariant Equation. For any $q \in QI$ and $s \in SA$, the following equation is called *Zero-invariant* equation (its left side is called Zero-invariant):

$$P(q, s, b) = 0, \text{ if either } q \in QI(b) \text{ or } s \in SA(b) \text{ is false.} \quad (6)$$

For example, in Figure 1(c), q_1 does not appear in the 3rd bucket, so we know $P(q_1, s, 3) = 0$ for any $s \in SA$. Similarly, s_1 does not appear in the 3rd bucket either, so we have $P(q, s_1, 3) = 0$ for any $q \in QI$.

Soundness. In the following theorem, we prove that all the above equations are indeed invariant equations; namely, they hold true for all the assignments in Bucket b.

THEOREM 1. *(Soundness) The above QI-invariant, SA-invariant, and Zero-invariant equations are sound.*

PROOF. Because of the following relationship,

$$1 = \sum_{j=1}^{h} P(s_j \mid q, b) = \sum_{j=1}^{h} \frac{P(q, s_j, b)}{P(q, b)},$$

we can immediately derive the QI-invariant equation in Eq. (4). The SA-invariant can be similarly derived.

For the Zero-invariant equations, if either q or s does not appear in Bucket b, the combination of (q, s, b) will never appear in any assignment. Therefore, $P(q, s, b) = 0$ is always true for all assignments in Bucket b. □

5.3 Completeness

Being able to find invariants is not enough, we must find all of them from the published data D'. Recall that invariants are the "absolute truth" about the original data; the more "truth" we collect from D', the more knowledge we will have on the original data, and the more accurately we can measure privacy. Therefore, to ensure the accuracy of privacy measure, we cannot miss any invariant from D'. This is the *completeness* property.

We call the QI-, SA-, and Zero- invariants the *base invariants*. Let U be the union of all these invariants. We call U the *invariant set*. We have the following theorem:

THEOREM 2. *(Completeness) The invariant set U is complete. Namely, for any probability expression F, F is an invariant, if and only if it can be constructed using the linear combination of the base invariants in U.*

PROOF. The "if" part of the theorem is trivial to prove, because a linear combination of invariants is still an invariant. We only prove the "only if" part. That is, we prove that

if F is an invariant, it can be constructed using the linear combination of the base invariants in U.

Before we prove the theorem, we make two simplifications. First, we replace F's probability terms that are Zero-invariants with zeros; this way, we only need to prove the theorem for a reduced U that only consists of QI-invariants and SA-invariants. Second, according to Lemma 1, we only need to prove this theorem for a single bucket (say Bucket b). Namely, we assume that F is a probability expression consisting of the probability terms from Bucket b, and that U consists of the invariants from the bucket b. Using the same notations from the last subsection, we let $QI(b) = \{q_1, \cdots, q_g\}$ and $SA(b) = \{s_1, \cdots, s_h\}$ to represent the sets of distinct QI values and SA values in Bucket b, respectively.

We use the contradiction approach by assuming that F is an invariant and F is not a linear combination of the base invariants. Our proof consists of two steps. First, we remove all the occurrences of q_1 and s_1 from F by using the linear transformation based on the base invariants. We use F^* to represent the resultant expression. Because F is an invariant, F^* is also an invariant. Second, we construct two assignments Λ_A and Λ_B; we show that $F^*(\Lambda_A) \neq F^*(\Lambda_B)$, which disproves that F^* is an invariant.

Step 1: Removing q_1 and s_1 from F. By rewriting the QI- and SA- invariants from Equations (4) and (5), we have

$$P(q, s_1, b) = P(q, b) - \sum_{j=2}^{h} P(q, s_j, b), \text{ for } q \in QI(b), \quad (7)$$

$$P(q_1, s, b) = P(s, b) - \sum_{i=2}^{g} P(q_i, s, b), \text{ for } s \in SA(b). \quad (8)$$

After we remove the zero-invariants from F, the probability terms in F that contain q_1 or s_1 must be either $P(q, s_1, b)$ for $q \in QI(b)$, or $P(q_1, s, b)$, for $s \in SA(b)$. Therefore, we can get rid of q_1 and s_1 from F by replacing the occurrences of q_1 and s_1 using Equations (7) and (8). We call the resultant expression F^*.

To help readers understand this step, we use a matrix to represent our SA- and QI- invariants. We call this matrix the *constraint matrix*. Each row of the constraint matrix represents an invariant, and each entry represents the coefficient of its corresponding probability term. Figure 3 turns the QI-invariants and SA-invariants of the Bucket 1 in Figure 1(c) into an invariant matrix. We take the QI-invariant $C_1 = P(q_1, s_1) + P(q_1, s_2) + P(q_1, s_3)$ as an example; we can see it has three non-zero entry, each representing one of its probability terms.

We also list F and F^* in Figure 3, where F^* is obtained by a linear transformation on F, i.e., $F^* = F - a_1C_1 - a_2C_2 - a_3C_3 - (a_4 - a_1)C_5 - (a_7 - a_1)C_6$. We can see that F^* does not contain q_1 or s_1.

Step 2: Proving that F^* is not an invariant. All the operations in Step 1 are linear transformations, so, if F is not a linear combination of the QI-invariants and SA-invariants, F^* is not a linear combination of those invariants either. Therefore, there must be at least one non-zero probability term in F^*. Without the loss of generality, we let this non-zero probability term be $P(q_i, s_j)$, where $i \neq 1$ and $j \neq 1$.

		s_1			s_2			s_3		
		q_1	q_2	q_3	q_1	q_2	q_3	q_1	q_2	q_3
QI-invariant	C_1	1	0	0	1	0	0	1	0	0
	C_2	0	1	0	0	1	0	0	1	0
	C_3	0	0	1	0	0	1	0	0	1
SA-invariant	C_4	1	1	1	0	0	0	0	0	0
	C_5	0	0	0	1	1	1	0	0	0
	C_6	0	0	0	0	0	0	1	1	1
F		a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9
F^*		0	0	0	0	a'_5	a'_6	0	a'_8	a'_9

Figure 3: An example of invariant matrix

We construct the following two assignments:

$$\Lambda_A = \{(q_1, s_1), (q_i, s_j)\} \cup \Lambda_{rest},$$
$$\Lambda_B = \{(q_1, s_j), (q_i, s_1)\} \cup \Lambda_{rest},$$

where Λ_{rest} represent the rest of the assignment. Both assignments have the same Λ_{rest}. We let $P_A(q, s)$ and $P_B(q, s)$ represent the joint probabilities of q and s under these two different assignments (we omit the bucket index b).

Because of the similarity between the two assignments, we know that except for four scenarios, $P_A(q, s) = P_B(q, s)$, for $q \in QI(b)$ and $s \in SA(b)$; these four scenarios include $P_A(q_i, s_j) \neq P_B(q_i, s_j)$, $P_A(q_1, s_1) \neq P_B(q_1, s_1)$, $P_A(q_1, s_j) \neq P_B(q_1, s_j)$ and $P_A(q_i, s_1) \neq P_B(q_i, s_1)$. However, $P(q_1, s_j)$, $P(q_i, s_1)$, and $P(s_1, q_1)$ do not appear in F^* at all, because they contain either q_1 or s_1. On the other hand, $P(q_i, s_j)$ does appear in F^* (that is why we picked $P(q_i, s_j)$ at the first place). Therefore $P(q_i, s_j)$ is the only probability term that decides whether $F^*(\Lambda_A)$ equals $F^*(\Lambda_B)$ or not. Because $P_A(q_i, s_j) \neq P_B(q_i, s_j)$, we know that $F^*(\Lambda_A) \neq F^*(\Lambda_B)$, which means F^* is not an invariant, so F is not an invariant either. This conclusion is contradictory to our assumption at the beginning of the proof. □

5.4 Conciseness

Having proved that the invariants we have identified are complete, we would also like to know whether those invariants are minimal, i.e., whether there are redundant invariants. Although redundant invariants will not cause any problem in maximum entropy estimation, too much redundancy can incur unnecessary computation overhead. The following theorem eases our concerns over redundancy.

THEOREM 3. *(**Conciseness**) Let U be the invariant set for a bucket. If we remove any one invariant from U, the resulting set U' is minimal, i.e., if one more invariant is removed from U', the resulting set becomes incomplete.*

The proof of this theorem is given in Appendix A. Readers can easily verify this theorem using the invariant matrix in Figure 3. First, we can see that the vectors in the invariant matrix are not linearly independent. We can verify this by calculating $(C_1+C_2+C_3)-(C_4+C_5+C_6)$; the result is zero, indicating at least one vector is redundant. Let us remove one of the invariant (say C_4), we can verify that the resultant invariant matrix becomes linearly independent; so there is no more redundant vectors. The conciseness property indicates that the invariants we have identified are minimal except for one redundant invariant (in each bucket).

5.5 Optimization

For a large data set, the ME computation can be quite expensive, because of the number of constraints for both data and background knowledge can be large. In this subsection, we investigate how we can take advantage of the relationship between data and background knowledge to optimize our ME estimation. We will first study how ME can be optimized when there is no background knowledge.

LEMMA 2. *(Independence without background knowledge) When there is no background knowledge, the distribution of $P(Q, S, B)$ in Bucket B is independent from the distribution of $P(Q, S, B')$ in Bucket $B' \neq B$. In other words, if we change the distribution of $P(Q, S, B)$, the distribution of $P(Q, S, B')$ does not need to be changed.*

PROOF. This lemma is obvious because when there is no background knowledge, there is no additional constraint between $P(Q, S, B)$ and $P(Q, S, B')$, for $B' \neq B$. □

This independence property only holds when there is no background knowledge. When there is background knowledge, if we change the distribution in one bucket, the distributions in other buckets might have to be changed. For example, in Figure 1(c), assuming that we have the following background knowledge (note that there are two records with q_3 among the 10 records, so $P(q_3) = 2/10$):

$$\begin{aligned} P(s_3 \mid q_3) &= 0.5, \quad \text{so} \\ P(q_3, s_3) &= 0.5 * P(q_3) = 0.5 * 2/10 = 0.1. \end{aligned}$$

By formulating the above knowledge as an ME constraint, we have the following:

$$P(q_3, s_3) = P(q_3, s_3, 1) + P(q_3, s_3, 2) = 0.1.$$

We can see that the if we change the value of $P(q_3, s_3, 1)$, the value of $P(q_3, s_3, 2)$ has to be changed accordingly, because of the constants caused by the background knowledge.

Based on Lemma 2, we have the following theorem:

THEOREM 4. *When there is no background knowledge, the maximum entropy of the entire data set can be achieved by achieving the maximum entropy in each bucket.*

PROOF. Based on Lemma 2, the distribution of $P(Q, S, B)$ is independent from bucket to bucket. Therefore, for each bucket $B = b$, we can find the distribution of $P(Q, S, b)$ to achieve the maximum entropy in b. We put those distributions together, we will get a distribution of $P(Q, S, B)$ that maximizes the entropy in all the buckets. Because of the entropy of the entire data set is the sum of the entropies of all the buckets, the sum is also maximized when each of its independent elements is maximized. □

Theorem 4 indicates that, when there is no background knowledge, to achieve the global maximum entropy (i.e., on the entire data), we only need to achieve the local maximum entropy (i.e., for each bucket). Lemma 2 guarantees that all the local maximums can be achieved simultaneously. Therefore, when there is no background knowledge, we can just focus on each bucket one at a time to find its ME distribution. This will significantly reduce the computation cost than if we apply ME on the entire data set.

Unfortunately, Theorem 4 will not be true when there is background knowledge. As we have shown before, background knowledge actually serves as the constraints among the buckets. Because of these constraints, distributions of different buckets are not independent anymore; therefore, we might not be able to achieve the local maximum entropies in all the buckets simultaneously. This means, we cannot apply ME on the buckets separately anymore.

However, we show that even if there is background knowledge, using Lemma 2, we can still reduce the computation cost of ME. The reason why Lemma 2 does not hold anymore is that background knowledge introduces extra constraints among the buckets. However, these constraints might not affect all the buckets; for those that are not affected, their in-bucket distribution of $P(Q, S, b)$ is still independent from the others. We call these buckets *irrelevant buckets*.

DEFINITION 5.6. *(Irrelevant bucket) A bucket b is irrelevant to background knowledge if b does not appear in any non-zero probability term of background-knowledge constraints.*

PROPOSITION 1. *If a bucket is irrelevant to the background knowledge, the maximum entropy of the entire data is achieved only if the maximum entropy of this bucket is also achieved.*

Based on Proposition 1, we can first identify those irrelevant buckets, apply ME to those buckets one at a time to get their local maximum. Then we apply ME to all the buckets that are relevant to the background knowledge. If there are many irrelevant buckets, the overall computation cost of ME can be significantly reduced.

Directly Compute ME. When a bucket b is irrelevant to the background knowledge, the maximum entropy within this bucket can be achieved by letting $P(S \mid Q, b)$ to be a uniform distribution for each occurrence of S in b, i.e.,

$$P(S \mid Q, b) = \frac{\#\text{ of } S \text{ in Bucket } b}{N_b}, \text{ for } Q \in QI(b), \qquad (9)$$

where N_b is the number of records in Bucket b. This is how most of the existing work computes $P(S \mid Q, b)$. This formula offers a direct way to compute $P(S \mid Q)$ for buckets that are irrelevant to the background knowledge. We formally specify this property as the following consistency theorem:

THEOREM 5. *(**Consistency**) When Bucket b is irrelevant to the background knowledge, the joint distribution of $P(Q, S, b)$, for $Q \in QI$ and $S \in SA$, derived from Eq. (9) maximizes the entropy defined in Eq. (2).*

The proof of this theorem is given in Appendix B. "Consistency" indicates that this calculation is consistent with the ways how $P(S \mid Q, b)$ is computed in the existing work when background knowledge is not considered. In other words, although the existing work does not explicitly try to achieve the maximum entropy, the underlying uniform-distribution assumption does lead to the maximum entropy when background knowledge is not considered.

6. KNOWLEDGE ABOUT INDIVIDUALS

As we have mentioned before, background knowledge can be classified into two major categories: knowledge about data distributions and knowledge about individuals. The previous two sections demonstrate how to integrate the knowledge about data distributions in privacy quantification. Our method is not limited to that type of knowledge; it can be

ID	QI	SA	Bucket
$\{i_1, i_2, i_3\}$	q_1	$\{s_1, s_2, s_2, s_3\}$	1
$\{i_1, i_2, i_3\}$	q_1		
$\{i_4, i_5\}$	q_2		
$\{i_6, i_7\}$	q_3		
$\{i_1, i_2, i_3\}$	q_1	$\{s_1, s_3, s_4\}$	2
$\{i_6, i_7\}$	q_3		
$\{i_8\}$	q_4		
$\{i_4, i_5\}$	q_2	$\{s_2, s_4, s_5\}$	3
$\{i_9\}$	q_5		
$\{i_{10}\}$	q_6		

Figure 4: A bucketized data set with expanded anonymous ID field.

extended to knowledge about individuals. A complete study of this type of knowledge will be pursued in our future work. In this paper, we show how the knowledge about individuals can be modeled as ME constraints.

Since this type of knowledge involves individuals, in our ME constraints, some kind of identity information need to included. For example, for the background knowledge, such as "Alice does not have Cancer", we need to be able to include the identity "Alice" somehow in the constraints. If Alice's QI value Q is unique in the published data set, we can simply use Q to represent Alice's identity. This way, we can still formulate our constraints using $P(Q, S, B)$. Unfortunately, Q might not be unique, and several people might have the same QI value. Therefore, Q cannot be used as identity. If we still use $P(Q, S, B)$ in our constraints, we are not representing the knowledge about a specific individual.

To model the knowledge about individuals, we introduce an identifier field to the bucketized data set. Since this field has already been removed during anonymization, the identifiers we add back to the disguised data are not real IDs; they are just *pseudonyms*. We expand the example in Figure 1(c), and generate a new table with pseudonyms in Figure 4.

For a QI value that is unique in the data set, there is only one pseudonym associated with it. For example, q_4, q_5 and q_6 are unique quasi-identifiers, so they each have a unique pseudonym. During the linking attacks, if we know that Alice (whose QI value is q_4) is in the data set, we replace the pseudonym i_8 with Alice's real name.

For a QI value that is not unique in the data set (e.g. q_1), we associate multiple pseudonyms to such a QI value, i.e., one distinct pseudonym for each occurrence (we assume that each person has only one record in the data set). For example, in Figure 4, we associate $\{i_1, i_2, i_3\}$ to each of the three occurrences of q_1. If we know that Bob (with q_1) is in the data set, we can assign any one of the i_1, i_2, and i_3 to Bob, reflecting the fact that we know nothing about which of the occurrences belong to Bob.

There are many types of background knowledge about individuals. We provide a list of background knowledge that can be modeled using linear constraints. We only list linear equations, but as we discussed before, if we replace the equality symbol with inequality symbol, we can use the extended maximum entropy model to handle those inequality constraints. Our list is not meant to be exhaustive; more studies are needed to understand what types of knowledge of individuals can or cannot be expressed as linear constraints.

(1) Probabilistic knowledge about an individual and single SA value. For example, we might know that *"The probability that Alice (whose QI is q_1) has Breast Cancer (s_1) is 0.2"*. To model the knowledge stated in this example, we first assign the pseudonym i_1 to Alice. From the knowledge, we know that $P(s_1 \mid i_1, q_1) = 0.2$. Because $P(i_1, q_1, s_1) = P(s_1 \mid i_1, q_1) * P(i_1, q_1)$ and $P(i_1, q_1) = \frac{1}{N}$ (because each person appears only once in the data, and N is the total number of records), we have the following constraint (the 4th element in the probability terms represents bucket index):

$$P(i_1, q_1, s_1, 1) + P(i_1, q_1, s_1, 2) = 0.2 * P(i_1, q_1) = 0.2 * \frac{1}{N}.$$

(2) Probabilistic knowledge about an individual and multiple SA values. For example we might know that *"Alice (with q_1) has either Breast Cancer (s_1) or HIV (s_4)"*. Assigning i_1 to Alice. We have the following constraints:

$$\begin{aligned} & P(i_1, q_1, s_1, 1) + P(i_1, q_1, s_1, 2) + P(i_1, q_1, s_4, 2) \\ = \quad & P(i_1, q_1) = \frac{1}{N}. \end{aligned}$$

(3) Probabilistic knowledge about multiple individuals. For example, we might know that *"Two people among Alice (with q_1), Bob (with q_2), and Charlie (with q_5) have HIV (s_4)"*. Let us assign i_1 to Alice, i_4 to Bob, and i_9 to Charlie. We have the following constraints:

$$P(i_1, q_1, s_4, 2) + P(i_4, q_2, s_4, 3) + P(i_9, q_5, s_4, 3) = \frac{2}{N}.$$

In this type of knowledge, if the knowledge statement is changed from "two people" to "at least two people", we can change the equality sign to inequality. As we mentioned before, the extended maximum entropy modeling can deal with inequality constraints [11].

Deriving Invariants from Data. Because of the addition of the pseudonyms, the invariants derived from the published data set need to be modified accordingly. The derivation process is similar to what we have discussed in Section 5. We omit the details in this paper.

7. EVALUATION

We have implemented the described ME method using C++. In our implementation, we apply the method of *Lagrange multipliers* to convert the *constrained* optimization problem to an *unconstrained* optimization problem, which is then solved using LBFGS. We use Nocedal's LBFGS software [16]. According to Malouf's comparison [18], LBFGS is one of the most efficient methods in solving ME problems. The computer used in our evaluation is a Pentium M (3.0Ghz, dual core) with 4GB memory.

We use the *Adult* dataset from UCI Machine Learning Repository [2] in our evaluations. We use the *education* attribute among those 14 attributes as our SA attribute. This SA attribute has 16 different categorical values. We use eight quasi-identifier attributes in our experiments. The whole dataset contains 14210 records that are bucketized into 2842 buckets with five records in each bucket to satisfy 5-diversity. [3]

[2] ftp://ftp.ics.uci.edu/pub/machine-learning-databases

[3] To be able to achieve 5-diversity for this specific dataset, we use a similar approach as that in [17], i.e., the most frequent values of SA is not considered as sensitive, and is excluded when checking whether the dataset is 5-diversified.

7.1 Background Knowledge vs. Privacy

This paper is not intended to give a specific privacy quantification metrics; instead our goal is to provide the most essential building block to privacy quantification. This building block is to derive the linking between SA and QI attributes, i.e., $P(SA \mid QI)$. The accuracy of this derivation decides the accuracy of privacy quantification. Therefore, to evaluate the effectiveness of our method, we measure how accurate our estimation is.

Let $P^*(SA \mid QI)$ be the conditional probability between QI and SA attributes; this is the probability that we derive from the published dataset using the proposed ME method. Let $P(SA \mid QI)$ be the original probability, i.e., the probability directly computed using the original data set. The difference between $P^*(SA \mid QI)$ and $P(SA \mid QI)$ indicates how accurate our estimation is. We use a variation of Kullback-Leibler distance [12] to quantify this difference. We call this measure *Estimation Accuracy*. Although this measure is not a measure for privacy, its value is a major indicator of privacy.

$$\text{Estimation Accuracy} = \sum_{q \in QI} P(q) \cdot \sum_{s \in SA} P(s|q) \log \frac{P(s|q)}{P^*(s|q)},$$

where the sum over SA is the KL distance between $P(s|q)$ and $P^*(s|q)$ for a given q. We take the weighted average of the KL distance for all different q values (the weight is the probability of q in the data set).

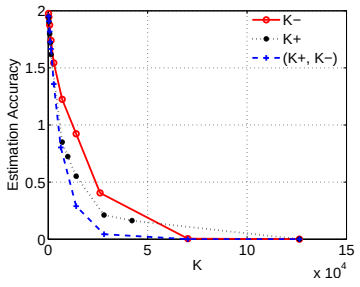


Figure 5: Positive and negative association rules

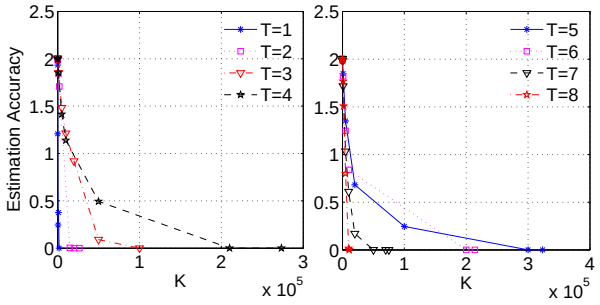


Figure 6: Number of QI attributes in knowledge

Amount of Background Knowledge. In this experiment, we measure how the amount of background knowledge affects privacy. We plot three curves. The K_- curve uses K negative association rules; the K_+ curve uses K positive association rules; the (K_+, K_-) curve uses $K/2$ positive association rules and $K/2$ negative association rules. The minimal support for these association rules is set to 3/14210, i.e., each association rule must be supported by at least three records. We plot the results in Figure 5.

From the figure, we clearly see that privacy becomes worse when more background knowledge is available. Privacy drops dramatically especially when K is small, and its dropping rate slows down when more and more background knowledge constraints are available. This is reasonable since the introduction of background knowledge brings extra information that does not exist in the published dataset, so the privacy becomes worse quickly when more background knowledge is available. However, when the amount of background knowledge becomes larger and larger, redundancy exists among the knowledge, so the effect on privacy also decreases.

Types of Background Knowledge. From Figure 5, we can also see the different change rates of privacy for the three types of background knowledge. Clearly, the curve for (K_+, K_-) drops the fastest; this indicates that even for the same amount of association rules, the mix of positive and negative association rules contains more information than positive or negative association rules alone. This is mainly because less redundancy exists in the mixture. The result reinforces our decision to use both types of association rules as the bound of background knowledge.

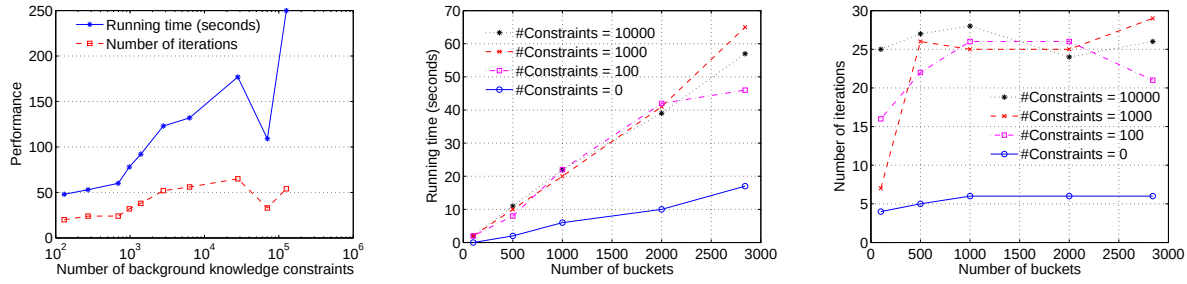
To understand how the number of QI attributes in the background knowledge affects privacy, we have conducted another experiment. We plot a curve for the background knowledge (association rules) that contains T QI attributes, where $T = 1, \ldots, 8$. The results are depicted in Figure 6. We have observed an interesting phenomena: the effect of background knowledge decreases when T changes from 1 to 4; then it swings back from 4 to 8. The decreases from $T = 1$ to 4 is intuitive, because the support of association rules for smaller T is usually larger; namely, a single rule can provide the background knowledge that affects more records. The increases from $T = 4$ to $T = 8$ is less intuitive. It is mainly because that after certain point, another effect begin to dominate: when T gets closer to 8, it provides more and more accurate information for the estimation of $P(SA \mid QI)$, where QI contains 8 attributes.

7.2 Performance

To understand how scalable our method is, we have conducted experiments to study how well our method works when the problem size increases. Since the ME method is an iterative method, we quantify performance in terms of both running time and number of iterations in the search process. In these experiments, we have not applied the optimization techniques discussed in Section 5.5.

We differentiate two types of knowledge and consider their influence on performance separately; they are the knowledge from the published dataset and the background knowledge. The nature of these two types of knowledge is different - a constraint derived from dataset only consists of the probability terms from the same bucket, while a constraint from background knowledge might involve the probability terms from multiple buckets. It is interesting to see how these two types of knowledge affect the performance of the ME estimation process.

Influence of the size of background knowledge. We fix the size of the dataset, so the number of constraints from



(a) Performance vs. Knowledge (b) Running time vs. Data Size (c) Iteration vs. Data Size

Figure 7: Performance versus Knowledge Size and Data Size

the dataset is fixed; we investigate the performance when the number of constraints from the background knowledge changes. The results are plotted in Figures 7(a), which shows that the running time and the number of iterations both increase when the number of constraints from background knowledge increases; however, the increase is quite slow, and only log-linear to the x-axis. There are some fluctuations in the curves. This is mainly because ME search processes are influenced by the constraints, and adding or removing some constraints may result in different search paths with different convergence speeds.

Influence of size of dataset. In this experiment, we fix the size of background knowledge and change the size of dataset, i.e., the number of buckets. The results are shown in Figures 7(b) and 7(c). In Figure 7(b), the legend indicates the number of constraints from the background knowledge.

Figure 7(b) shows that the running time increases almost linearly to the increment of bucket number. On the other hand, Figure 7(c) shows that the number of iterations is almost constant in most parts of the region. However, because each iteration now takes more time to compute when the bucket number increases, the running time actually increases. Both observations are consistent with the inherent characteristic of these two different types of knowledge that we have explained earlier.

8. CONCLUSION AND FUTURE WORK

We propose a systematic method to incorporate background knowledge in privacy quantification. Our method is based on the maximum entropy principle. In our method, we model the background knowledge as linear constraints; we also derive all the invariants from the published data, and represent the invariants as linear constraints. We then feed these constraints to an optimization procedure, which outputs the maximum entropy results that satisfy those constraints. These results are used for privacy quantification. We also propose a way to bound the amount of background knowledge, and have studied the privacy property of published data under various bounds.

The work presented in this paper opens up several directions for future work. One direction is to apply the similar method to other data disguising methods, such as generalization and randomization. The second direction is to study how the background knowledge about individuals can affect privacy. The third direction is to address other types of background knowledge, such as the knowledge that is represented as inequalities.

9. ACKNOWLEDGMENT

We thank Dr. Biao Chen for the discussion of maximum entropy modeling. We also thank the anonymous reviewers for their comments and kind suggestions. This work is supported in part by Grant CNS-0430252 from the US National Science Foundation and also by Grant W911NF-05-1-0247 from the US Army Research Office. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

10. REPEATABILITY ASSESSMENT RESULT

Figures 5 and 6 have been verified by the SIGMOD repeatability committee. Code used in the paper are available at http://www.sigmod.org/codearchive/sigmod2008/.

11. REFERENCES

[1] 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.

[2] R. Agrawal, T. Imielinski, and A. Swami. Mining association rules between sets of items in large databases. In P. Buneman and S. Jajodia, editors, *Proceedings of ACM SIGMOD Conference on Management of Data*, pages 207–216, Washington D.C., May 1993.

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

[4] 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.

[5] A. L. Berger, S. D. Pietra, and V. J. D. Pietra. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1), 1996.

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

[7] 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.

[8] J. N. Darroch and D. Ratcliff. Generalized iterative scaling for log-linear models. *Annals of Mathematicsl Statistics*, (32):1470–1480, 1872.

[9] 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.

[10] 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.

[11] J. Kazama and J. Tsujii. Maximum entropy models with inequality constraints: A case study on text categorization. *Machine Learning (Special Issue on Learning in Speech and Language Technologies)*, 60(1-3):159–194, September 2005.

[12] S. Kullback and R. A. Leibler. On information and sufficiency. pages 79–86, 1951.

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

[14] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Mondrian multidimensional k-anonymity. In *Proceedings of the 22nd IEEE International Conference on Data Engineering (ICDE)*, Atlanta, Georgia, USA, April 2006.

[15] N. Li, T. Li, and S. Venkatasubramanian. 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.

[16] D. Liu and J. Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming B*, 45:503–528, 1989.

[17] 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.

[18] R. Malouf. A comparison of algorithms for maximum entropy parameter estimation, 2002.

[19] 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.

[20] S. D. Pietra, V. D. Pietra, and J. Lafferty. Inducing features of random fields. *Transactions Pattern Analysis and Machine Intelligence*, 19(4), April 1997.

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

[22] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical report, 1998.

[23] A. Savasere, E. Omiecinski, and S. B. Navathe. Mining for strong negative associations in a large database of customer transactions. In *In Proceedings of the IEEE International Conference on Data Engineering (ICDE)*, pages 494–502, 1998.

[24] 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.

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

APPENDIX

A. PROOF OF CONCISENESS

PROOF. Suppose there are g and h distinctive QI and SA values respectively within one bucket, we need to prove that $g+h-1$ out of the total $g+h$ constraints for the disguised dataset are linearly independent. We use the contradiction approach by assuming that they are not linearly independent, i.e., we can find a linear combination of them to obtain value of zero. Without loss of generality, let's remove one constraint, the QI-invariant constraint related to Q_1. If the rest of the constraints are not linearly independent, we can find the following linear combination:

$$\sum_{i=2}^{g} A_i \left[\sum_{j=1}^{h} P(Q_i, S_j, b)\right] + \sum_{j=1}^{h} B_j \left[\sum_{i=1}^{g} P(Q_i, S_j, b)\right] = 0.$$

Each probability $P(Q_i, S_j, b)$ shows up once in the first summation and once in the second summation. Consider $P(Q_1, S_j, b)$: it only shows up in the second summation once, so its corresponding coefficient must be zero, i.e., $B_j = 0$. Consider $P(Q_i, S_j, b)$, for $i > 1$: it only shows up once in the first summation (the only remaining term), so its coefficient must be zero too, i.e., $A_i = 0$. Therefore, these constraints are linearly independent from each other.

On the other hand, if we do not remove any constraint, we can get the following:

$$\sum_{i=1}^{g} \left[\sum_{j=1}^{h} P(Q_i, S_j, b)\right] - \sum_{j=1}^{h} \left[\sum_{i=1}^{g} P(Q_i, S_j, b)\right] = 0.$$

Therefore, these constraints are not linearly independent. □

B. PROOF OF CONSISTENCY

PROOF. Maximum entropy can be achieved if each bucket achieves its own maximum entropy, when conditional probabilities of one bucket cannot affect conditional probabilities in other buckets. This is true in our case when we have no background knowledge. Therefore, our proof is narrowed

down to prove that the solution we get in each bucket is actually the maximum entropy result.

We focus on bucket b. Suppose this bucket has N_b records, and there are g and h distinctive QI and SA values respectively. The proof is mathematically straightforward, i.e., we simply solve the optimization problem using *Lagrange multipliers* analytically.

The unconditional optimization problem aims to maximize the following expression:

$$\begin{aligned} L_b &= -\sum_{Q,S} P(Q,S,b)\log P(Q,S,b) \\ &\quad + \sum_{i=1}^{g} A_i[\sum_{j=1}^{h} P(Q_i,S_j,b) - P(Q_i,b)] \\ &\quad + \sum_{j=1}^{h} B_j[\sum_{i=1}^{g} P(Q_i,S_j,b) - P(S_j,b)], \end{aligned}$$

where A_i and B_j are *Lagrange multipliers*.

For all values of Q_i and S_j that show up in the bucket, the joint probability $P(Q_i,S_j,b)$ also show up once in each of the three terms in Equation (10). Taking partial derivative over $P(Q_i,S_j,b)$, we have

$$-\log P(Q_i,S_j,b) - 1/\ln 2 + A_i + B_j = 0$$

So

$$P(Q_i,S_j,b) = 2^{A_i+B_j-1/\ln 2} \tag{10}$$

Plugging Equation (10) back into QI-invariant with *Lagrange multiplier* A_i, we have

$$2^{A_i-1/\ln 2}\sum_{j=1}^{h} 2^{B_j} = P(Q_i,b) \tag{11}$$

Plugging Equation (10) back into SA-invariant with *Lagrange multiplier* B_j, we have

$$2^{B_j-1/\ln 2}\sum_{i=1}^{g} 2^{A_i} = P(S_j,b) \tag{12}$$

For any two values i_1, i_2 assigned to i in Equation (11), we have

$$\frac{2^{A_{i_1}}}{2^{A_{i_2}}} = \frac{P(Q_{i_1},b)}{P(Q_{i_2},b)} \tag{13}$$

For any two values j_1, j_2 assigned to j in Equation (12), we have

$$\frac{2^{B_{j_1}}}{2^{B_{j_2}}} = \frac{P(S_{j_1},b)}{P(S_{j_2},b)} \tag{14}$$

Plugging Equations (13) and (14) back into Equation (10), we have

$$\frac{P(Q_{i_1},S_{j_1},b)}{P(Q_{i_2},S_{j_2},b)} = \frac{P(Q_{i_1},b)P(S_{j_1},b)}{P(Q_{i_2},b)P(S_{j_2},b)}$$

Letting $i_1 = i_2$, we have

$$\frac{P(Q_i,S_{j_1},b)}{P(Q_i,S_{j_2},b)} = \frac{P(S_{j_1},b)}{P(S_{j_2},b)}$$

So,

$$P(Q_i,S_{j_1},b) = P(Q_i,S_{j_2},b)\frac{P(S_{j_1},b)}{P(S_{j_2},b)} \tag{15}$$

We also know

$$\sum_{j_1} P(Q_i,S_{j_1},b) = P(Q_i,b) \tag{16}$$

Plugging Equation (15) into Equation (16), we have

$$P(Q_i,S_j,b) = \frac{P(Q_i,b)P(S_j,b)}{P(b)}$$

If there is indeed only one bucket, we know $P(b) = 1$, so

$$P(Q_i,S_j,b) = P(Q_i,b)P(S_j,b) \tag{17}$$

The Equation (17) essentially says that QIs and SAs are independent, as a result of the broken linkage between QI and SA values. □