ℓ-Diversity: Privacy Beyond k-Anonymity

Ashwin Machanavajjhala Daniel Kifer Johannes Gehrke
Muthuramakrishnan Venkitasubramaniam
Department of Computer Science, Cornell University
{mvnak, dkifer, johannes, vmuthu}@cs.cornell.edu

Abstract

Publishing data about individuals without revealing sensitive information about them is an important problem. In recent years, a new definition of privacy called k-anonymity has gained popularity. In a k-anonymized dataset, each record is indistinguishable from at least $k-1$ other records with respect to certain "identifying" attributes.

In this paper we show using two simple attacks that a k-anonymized dataset has some subtle, but severe privacy problems. First, an attacker can discover the values of sensitive attributes when there is little diversity in those sensitive attributes. This kind of attack is a known problem [60]. Second, attackers often have background knowledge, and we show that k-anonymity does not guarantee privacy against attackers using background knowledge. We give a detailed analysis of these two attacks and we propose a novel and powerful privacy criterion called ℓ-diversity that can defend against such attacks. In addition to building a formal foundation for ℓ-diversity, we show in an experimental evaluation that ℓ-diversity is practical and can be implemented efficiently.

1. Introduction

Many organizations are increasingly publishing microdata – tables that contain unaggregated information about individuals. These tables can include medical, voter registration, census, and customer data. Microdata is a valuable source of information for the allocation of public funds, medical research, and trend analysis. However, if individuals can be uniquely identified in the microdata, then their private information (such as their medical condition) would be disclosed, and this is unacceptable.

To avoid the identification of records in microdata, uniquely identifying information like names and social security numbers are removed from the table. However, this first sanitization still does not ensure the privacy of individuals in the data. A recent study estimated that 87% of the population of the United States can be uniquely identified using the seemingly innocuous attributes gender, date of birth, and 5-digit zip code [67]. In fact, those three attributes were used to link Massachusetts voter registration records (which included the name, gender, zip code, and date of birth) to supposedly anonymized medical data from GIC[1] (which included gender, zip code, date of birth and diagnosis). This "linking attack" managed to uniquely identify the medical records of the governor of Massachusetts in the medical data [68].

Sets of attributes (like gender, date of birth, and zip code in the example above) that can be linked with external data to uniquely identify individuals in the population are called *quasi-identifiers*. To counter linking attacks using quasi-identifiers, Samarati and Sweeney proposed a definition of privacy called *k-anonymity* [62, 68]. A table satisfies k-anonymity if every record in the table is indistinguishable from at least $k-1$ other records with respect to every set of quasi-identifier attributes; such a table is called a *k-anonymous* table. Hence, for every combination of values of the quasi-identifiers in the k-anonymous table, there are at least k records that share those values. This ensures that individuals cannot be uniquely identified by linking attacks.

An Example. Figure 1 shows medical records from a fictitious hospital located in upstate New York. Note that the table contains no uniquely identifying attributes like name, social security number, etc. In this example, we divide the attributes

[1] Group Insurance Company (GIC) is responsible for purchasing health insurance for Massachusetts state employees.

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	13053	28	Russian	Heart Disease
2	13068	29	American	Heart Disease
3	13068	21	Japanese	Viral Infection
4	13053	23	American	Viral Infection
5	14853	50	Indian	Cancer
6	14853	55	Russian	Heart Disease
7	14850	47	American	Viral Infection
8	14850	49	American	Viral Infection
9	13053	31	American	Cancer
10	13053	37	Indian	Cancer
11	13068	36	Japanese	Cancer
12	13068	35	American	Cancer

Figure 1. Inpatient Microdata

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	130**	< 30	$*$	Heart Disease
2	130**	< 30	$*$	Heart Disease
3	130**	< 30	$*$	Viral Infection
4	130**	< 30	$*$	Viral Infection
5	1485*	≥ 40	$*$	Cancer
6	1485*	≥ 40	$*$	Heart Disease
7	1485*	≥ 40	$*$	Viral Infection
8	1485*	≥ 40	$*$	Viral Infection
9	130**	$3*$	$*$	Cancer
10	130**	$3*$	$*$	Cancer
11	130**	$3*$	$*$	Cancer
12	130**	$3*$	$*$	Cancer

Figure 2. 4-anonymous Inpatient Microdata

into two groups: the *sensitive* attributes (consisting only of medical condition) and the *non-sensitive* attributes (zip code, age, and nationality). An attribute is marked sensitive if an adversary must not be allowed to discover the value of that attribute for any individual in the dataset. Attributes not marked sensitive are non-sensitive. Furthermore, let the collection of attributes $\{$zip code, age, nationality$\}$ be the quasi-identifier for this dataset. Figure 2 shows a 4-anonymous table derived from the table in Figure 1 (here "*" denotes a suppressed value so, for example, "zip code = 1485*" means that the zip code is in the range $[14850 - 14859]$ and "age=3*" means the age is in the range $[30 - 39]$). Note that in the 4-anonymous table, each tuple has the same values for the quasi-identifier as at least three other tuples in the table.

Because of its conceptual simplicity, k-anonymity has been widely discussed as a viable definition of privacy in data publishing, and due to algorithmic advances in creating k-anonymous versions of a dataset [3, 12, 51, 57, 62, 68, 74], k-anonymity has grown in popularity. However, does k-anonymity really guarantee privacy? In the next section, we will show that the answer to this question is interestingly *no*. We give examples of two simple, yet subtle attacks on a k-anonymous dataset that allow an attacker to identify individual records. Defending against these attacks requires a stronger notion of privacy that we call ℓ-diversity, the focus of this paper. But we are jumping ahead in our story. Let us first show the two attacks to give the intuition behind the problems with k-anonymity.

1.1. Attacks On k-Anonymity

In this section we present two attacks, the *homogeneity attack* and the *background knowledge attack*, and we show how they can be used to compromise a k-anonymous dataset.

Homogeneity Attack: Alice and Bob are antagonistic neighbors. One day Bob falls ill and is taken by ambulance to the hospital. Having seen the ambulance, Alice sets out to discover what disease Bob is suffering from. Alice discovers the 4-anonymous table of current inpatient records published by the hospital (Figure 2), and so she knows that one of the records in this table contains Bob's data. Since Alice is Bob's neighbor, she knows that Bob is a 31-year-old American male who lives in the zip code 13053 (the quiet town of Dryden). Therefore, Alice knows that Bob's record number is 9, 10, 11, or 12. Now, all of those patients have the same medical condition (cancer), and so Alice concludes that Bob has cancer.

Observation 1. *k-Anonymity can create groups that leak information due to lack of diversity in the sensitive attribute.*

Such a situation is not uncommon. As a back-of-the-envelope calculation, suppose we have a dataset containing 60,000 distinct tuples where the sensitive attribute can take three distinct values and is not correlated with the non-sensitive attributes. A 5-anonymization of this table will have around 12,000 groups[2] and, on average, 1 out of every 81 groups will have no diversity (the values for the sensitive attribute will all be the same). Thus we should expect about 148 groups with no diversity. Therefore, information about 740 people would be compromised by a homogeneity attack. This suggests that in addition to k-anonymity, the sanitized table should also ensure "diversity" – all tuples that share the same values of their quasi-identifiers should have diverse values for their sensitive attributes.

[2]Our experiments on real data sets show that data is often very skewed and a 5-anonymous table might not have so many groups

The possibility of a homogeneity attack has been previously discussed in the literature (e.g., [60]). One solution to the homogeneity problem, as presented by Ohrn et al. [60], turns out to be a specific instance of our general principle of ℓ-diversity (see Section 4). For reasons that will become clear in Section 4, we refer to that method as *entropy ℓ-diversity*. By examining privacy from a different perspective, we prove additional privacy-preserving properties of entropy ℓ-diversity. We also present other privacy definitions that satisfy the principle of ℓ-diversity that have greater flexibility.

The next observation is that an adversary could use "background" knowledge to discover sensitive information.

Background Knowledge Attack: Alice has a pen-friend named Umeko who is admitted to the same hospital as Bob, and whose patient records also appear in the table shown in Figure 2. Alice knows that Umeko is a 21 year-old Japanese female who currently lives in zip code 13068. Based on this information, Alice learns that Umeko's information is contained in record number 1,2,3, or 4. Without additional information, Alice is not sure whether Umeko caught a virus or has heart disease. However, it is well-known that Japanese have an extremely low incidence of heart disease. Therefore Alice concludes with near certainty that Umeko has a viral infection.

Observation 2. *k-Anonymity does not protect against attacks based on background knowledge.*

We have demonstrated (using the homogeneity and background knowledge attacks) that a k-anonymous table may disclose sensitive information. Since both of these attacks are plausible in real life, we need a stronger definition of privacy that takes into account diversity and background knowledge. This paper addresses this very issue.

1.2. Contributions and Paper Outline

In the previous section, we showed that k-anonymity is susceptible to homogeneity and background knowledge attacks; thus a stronger definition of privacy is needed. In the remainder of the paper, we derive our solution. We start by introducing an ideal notion of privacy called *Bayes-optimal* for the case that both data publisher and the adversary have knowledge of the complete joint distribution of the sensitive and nonsensitive attributes (Section 3). Unfortunately in practice, the data publisher is unlikely to possess all this information, and in addition, the adversary may have more specific background knowledge than the data publisher. Hence, while Bayes-optimal privacy sounds great in theory, it is unlikely that it can be guaranteed in practice. To address this problem, we show that the notion of Bayes-optimal privacy naturally leads to a novel *practical* criterion that we call ℓ-diversity. ℓ-Diversity provides privacy even when the data publisher does not know what kind of knowledge is possessed by the adversary. The main idea behind ℓ-diversity is the requirement that the values of the sensitive attributes are well-represented in each group (Section 4).

We show that existing algorithms for k-anonymity can be adapted to compute ℓ-diverse tables (Section 5), and in an experimental evaluation we show that ℓ-diversity is practical and can be implemented efficiently (Section 6). We discuss related work in Section 7, and we conclude in Section 8. Before jumping into the contributions of this paper, we introduce the notation needed to formally discuss data privacy in the next section.

2. Model and Notation

In this section we will introduce some basic notation that will be used in the remainder of the paper. We will also discuss how a table can be anonymized and what kind of background knowledge an adversary may possess.

Basic Notation. Let $T = \{t_1, t_2, \ldots, t_n\}$ be a table with attributes $A_1, \ldots, A_m$. We assume that T is a subset of some larger population Ω where each tuple $t_i \in T$ represents an individual from the population. For example, if T is a medical dataset then Ω could be the population of the Caribbean island, San Lorenzo. Let $\mathcal{A}$ denote the set of all attributes $\{A_1, A_2, \ldots, A_m\}$ and $t[A_i]$ denote the value of attribute A_i for tuple t. If $\mathcal{C} = \{C_1, C_2, \ldots, C_p\} \subseteq \mathcal{A}$ then we use the notation $t[\mathcal{C}]$ to denote the tuple $(t[C_1], \ldots, t[C_p])$, which is the projection of t onto the attributes in $\mathcal{C}$.

In privacy-preserving data publishing, there exist several important subsets of $\mathcal{A}$. A *sensitive attribute* is an attribute whose value for any particular individual must be kept secret from people who have no direct access to the original data. Let $\mathcal{S}$ denote the set of all sensitive attributes. An example of a sensitive attribute is *Medical Condition* from Figure 1. The association between individuals and *Medical Condition* should be kept secret; thus we should not disclose which particular patients have cancer, but it is permissible to disclose the information that there exist cancer patients in the hospital. We assume that the data publisher knows which attributes are sensitive. To simplify the discussion, for much of this paper we will also assume that there is only one sensitive attribute; the extension of our results to multiple sensitive attributes is not difficult and is handled in Section 4.3. All attributes that are not sensitive are called *nonsensitive* attributes. Let $\mathcal{N}$ denote the set of nonsensitive attributes. We are now ready to formally define the notion of a quasi-identifier.

Definition 2.1 (Quasi-identifier)**.** *A set of nonsensitive attributes $\{Q_1, \ldots, Q_w\}$ of a table is called a* quasi-identifier *if these attributes can be linked with external data to uniquely identify at least one individual in the general population Ω.*

One example of a quasi-identifier is a primary key like social security number. Another example is the set {Gender, Age, Zip Code} in the GIC dataset that was used to identify the governor of Massachusetts as described in the introduction. Let us denote the set of all quasi-identifiers by $\mathcal{QI}$. We are now ready to formally define k-anonymity.

Definition 2.2 (k-Anonymity)**.** *A table T satisfies k-anonymity if for every tuple $t \in T$ there exist $k-1$ other tuples $t_{i_1}, t_{i_2}, \ldots, t_{i_{k-1}} \in T$ such that $t[\mathcal{C}] = t_{i_1}[\mathcal{C}] = t_{i_2}[\mathcal{C}] = \cdots = t_{i_{k-1}}[\mathcal{C}]$ for all $\mathcal{C} \in \mathcal{QI}$.*

The Anonymized Table $T^\star$. Since the quasi-identifiers might uniquely identify tuples in T, the table T is not published; it is subjected to an *anonymization procedure* and the resulting table $T^\star$ is published instead.

There has been a lot of research on techniques for anonymization (see Section 7 for a discussion of related work). These techniques can be broadly classified into *generalization* techniques [3, 51], *generalization with tuple suppression* techniques [12, 63], and *data swapping and randomization* techniques [1, 40]. In this paper we limit our discussion only to generalization techniques.

Definition 2.3 (Domain Generalization)**.** *A domain $D^\star = \{P_1, P_2, \ldots\}$ is a* generalization *(partition) of a domain D if $\bigcup P_i = D$ and $P_i \cap P_j = \emptyset$ whenever $i \neq j$. For $x \in D$ we let $\phi_{D^\star}(x)$ denote the element $P \in D^\star$ that contains x.*

Note that we can create a partial order $\prec_G$ on domains by requiring $D \prec_G D^\star$ if and only if $D^\star$ is a generalization of D. Given a table $T = \{t_1, \ldots, t_n\}$ with the set of nonsensitive attributes $\mathcal{N}$ and a generalization $D^\star_N$ of domain($\mathcal{N}$), we can construct a table $T^\star = \{t_1^\star, \ldots, t_n^\star\}$ by replacing the value of $t_i[N]$ with the generalized value $\phi_{D^\star_N}(t_i[N])$ to get a new tuple $t_i^\star$. The tuple $t_i^\star$ is called a *generalization* of the tuple t_i and we use the notation $t_i \xrightarrow{\star} t_i^\star$ to mean "$t_i^\star$ generalizes t_i". Extending the notation to tables, $T \xrightarrow{\star} T^\star$ means "$T^\star$ is a generalization of T". Typically, ordered attributes are partitioned into intervals, and categorical attributes are partitioned according to a user-defined hierarchy (for example, cities are generalized to counties, counties to states, and states to regions).

Example 1 (Continued). The table in Figure 2 is a generalization of the table in Figure 1. We generalized on the *Zip Code* attribute by partitioning it into two sets: "1485*" (representing all zip codes whose first four digits are 1485) and "130**" (representing all zip codes whose first three digits are 130). Then we partitioned *Age* into three groups: "< 30", "3*" (representing all ages between 30 and 39), and "≥ 40". Finally, we partitioned *Nationality* into just one set "*" representing all nationalities.

The Adversary's Background Knowledge. Since the background knowledge attack was due to the adversary's additional knowledge about the table, let us briefly discuss the type of background knowledge that we are modeling.

First, the adversary has access to the published table $T^\star$ and she knows that $T^\star$ is a generalization of some base table T. The adversary also knows the domain of each attribute of T.

Second, the adversary may know that some individuals are in the table. This knowledge is often easy to acquire. For example, GIC published medical data about all Massachusetts state employees. If the adversary Alice knows that her neighbor Bob is a Massachusetts state employee then Alice is almost certain that Bob's information is contained in that table. In this case, we assume that Alice knows all of Bob's nonsensitive attributes. In addition, the adversary could have knowledge about the sensitive attributes of specific individuals in the population and/or the table. For example, the adversary Alice might know that neighbor Bob does not have pneumonia since Bob does not show any of the symptoms of pneumonia. We call such knowledge "instance-level background knowledge," since it is associated with specific instances in the table. In addition, Alice may know complete information about some people in the table other than Bob (for example, Alice's data may be in the table).

Third, the adversary could have partial knowledge about the distribution of sensitive and nonsensitive attributes in the population. We call this "demographic background knowledge." For example, the adversary may know
$P\left(t[\text{Condition}] = \text{"cancer"} \middle| t[\text{Age}] \geq 40\right)$ and may use it to make additional inferences about records in the table.

Now armed with the right notation, let us start looking into principles and definitions of privacy that leak little information.

3. Bayes-Optimal Privacy

In this section we analyze an ideal notion of privacy. We call it *Bayes-Optimal Privacy* since it involves modeling background knowledge as a probability distribution over the attributes and uses Bayesian inference techniques to reason about

privacy. We introduce tools for reasoning about privacy (Section 3.1), use them to discuss theoretical principles of privacy (Section 3.2), and then point out the difficulties that need to be overcome to arrive at a practical definition of privacy (Section 3.3).

3.1. Changes in Belief Due to Data Publishing

For simplicity of discussion, we combine all the nonsensitive attributes into a single, multi-dimensional quasi-identifier attribute Q whose values are generalized to create the anonymized table $T^\star$ from the base table T. Since Bayes-optimal privacy is only used to motivate a practical definition, we make the following two simplifying assumptions: first, we assume that T is a simple random sample from some larger population Ω (a sample of size n drawn without replacement is called a *simple random sample* if every sample of size n is equally likely); second, we assume that there is a single sensitive attribute. We would like to emphasize that both these assumptions will be dropped in Section 4 when we introduce a practical definition of privacy.

Recall that in our attack model, the adversary Alice has partial knowledge of the distribution of the sensitive and non-sensitive attributes. Let us assume a worst case scenario where Alice knows the complete joint distribution f of Q and S (i.e., she knows their frequency in the population Ω). Consider any individual Bob that Alice knows is in the table. She knows that Bob corresponds to a record $t \in T$ that has been generalized to a record t^* in the published table $T^\star$. She also knows the value of Bob's non-sensitive attributes (i.e., she knows that $t[Q] = q$). Alice's goal is to use her background knowledge to discover Bob's sensitive information — the value of $t[S]$. We gauge her success using two quantities: Alice's *prior belief*, and her *posterior belief*.

Alice's *prior belief*, $\alpha_{(q,s)}$, that Bob's sensitive attribute is s given that his nonsensitive attribute is q, is just her background knowledge:

$$\alpha_{(q,s)} = P_f\left(t[S] = s \middle| \, t[Q] = q\right)$$

After Alice observes the table $T^\star$, her belief about Bob's sensitive attribute changes. This new belief, $\beta_{(q,s,T^\star)}$, is her *posterior belief*:

$$\beta_{(q,s,T^\star)} = P_f\left(t[S] = s \middle| \, t[Q] = q \wedge \exists t^\star \in T^\star,\, t \xrightarrow{\star} t^\star\right)$$

Given f and $T^\star$, we can derive a formula for $\beta_{(q,s,T^\star)}$ which will help us formulate our new privacy definition in Section 4. The main idea behind the derivation is to find a set of equally likely disjoint random worlds (like in [11]) such that a conditional probability $P(A|B)$ is the number of worlds satisfying the condition $A \wedge B$ divided by the number of worlds satisfying the condition B.

Theorem 3.1. *Let $T^\star$ be a published table which is obtained by performing generalizations on a table T; let X be an individual with $X[Q] = q$ who appears in the table T (and also $T^\star$); let $q^\star$ be the generalized value of q in $T^\star$; let s be a possible value of the sensitive attribute; let $n_{(q^\star,s')}$ be the number of tuples $t^\star \in T^\star$ where $t^\star[Q] = q^\star$ and $t^\star[S] = s'$; and let $f(s' \mid q^\star)$ be the conditional probability of the sensitive attribute being s' conditioned on the fact that the nonsensitive attribute Q is some q' which can be generalized to $q^\star$. Then the observed belief that $X[S] = s$ is given by:*

$$\beta_{(q,s,T^\star)} \quad = \quad \frac{n_{(q^\star,s)} \frac{f(s|q)}{f(s|q^\star)}}{\sum_{s' \in S} n_{(q^\star,s')} \frac{f(s'|q)}{f(s'|q^\star)}} \tag{1}$$

Proof. For ease of reference, we review the notation used in this proof in Figure 3.

To help us model the adversary's uncertainty about the value of Bob's sensitive attribute after seeing the anonymized table $T^\star$, we will construct a set of *random worlds* such that $T^\star$ could have come from any one of these random worlds with equal probability. In all of these worlds, Bob (or X, as we will call him in this proof) appears in $T^\star$. In any two different random worlds, either some individual in the population has a different value for the sensitive attribute, or a different set of individuals appear in $T^\star$. Since the random worlds are equally likely and mutually exclusive, the required conditional probability is the fraction of the total number of worlds in which $X[S] = s$ (as in [11]).

Constructing the set of random worlds:

Formally, a random world is a pair (ψ, Z_n) where $\psi : \Omega \to S$ is an assignment of sensitive values for each individual $\omega \in \Omega$ and Z_n is a simple random sample of n individuals from Ω. We are interested in only those assignments ψ which are consistent with the adversary's background knowledge. In particular, the adversary knows the size of Ω and the distribution of

Notation	Description
T	Un-anonymized table
$T^\star$	The anonymized table
Q	Domain of the quasi-identifier attribute
$Q^\star$	Generalized domain of the quasi-identifier attribute
S	Domain of the sensitive attribute
Ω	Population of individuals
X	Bob, the individual in the population Ω with $X[Q] = q$ and who is known to be in T
N_q	Number of individuals w in the population Ω such that $w[Q] = q$
$N_{(q,s)}$	Number of individuals w in the population Ω such that $w[Q] = q$ and $w[S] = s$
$N_{(q^\star,s)}$	Number of individuals w in the population Ω such that $w[S] = s$ and $w[Q^\star] = q^\star$
n	Number of tuples in the anonymized table $T^\star$
$n_{(q^\star,s)}$	Number of tuples $t^\star$ in the anonymized table $T^\star$ such that $t^\star[S] = s$ and $t^\star[Q^\star] = q^\star$

Figure 3. Notation used in the Proof of Theorem 3.1

sensitive and nonsensitive attributes; in other words, for every (q, s), the adversary knows $N_{(q,s)}$ – the number of individuals with nonsensitive attribute q who have sensitive value s. Therefore for every (q, s), ψ should assign the value s to exactly $N_{(q,s)}$ out of the N_q individuals who have the nonsensitive value q. Note that in any two distinct assignments ψ_1, ψ_2 there is some individual ω such that $\psi_1(\omega) \neq \psi_2(\omega)$; i.e., ω is assigned to different values of S. Moreover, given only knowledge of the distribution of sensitive and nonsensitive attributes, the adversary has no preference for any of the ψ and, invoking the principle of indifference, considers each ψ to be equally likely.

The second component of a random world is Z_n. Z_n is a size n simple random sample from the population Ω. By the definition of a simple random sample, each Z_n is equally likely. Since the sample Z_n is picked independent of the assignment ψ, each random world (ψ, Z_n) is equally likely.

Each (ψ, Z_n) describes a table $T_{(\psi,Z_n)}$ containing n tuples with Q and S as attributes. We are interested in only those random worlds where X appears in $T_{(\psi,Z_n)}$ and where $T_{(\psi,Z_n)} \rightarrow^\star T^\star$. We can rephrase this condition as follows. We say that a random world (ψ, Z_n) is *compatible* with the published table $T^\star$ containing X, written as $(\psi, Z_n) \vdash (T^\star, X)$, if the following two conditions hold:

- $X \in Z_n$, where X is the individual with $X[Q] = q$ who is known to be in the table; and
- for every $(q^\star, s)$ pair there are $n_{(q^\star,s)}$ individuals ω in Z_n such that $\omega[Q]$ is generalized to $q^\star$ and such that $\psi(\omega) = s$.

The set of compatible random worlds completely characterizes the set of worlds which give rise to the anonymized table $T^\star$ containing X. It is clear that these worlds are equally likely. Also any two compatible random worlds are mutually exclusive because either some individual in the population is assigned a different value for S or the sample of individuals Z_n is different.

Calculating the conditional probability $\beta_{(q,s,T^\star)}$:

To calculate the conditional probability $\beta_{(q,s,T^\star)}$, we need to find the fraction of the total number of compatible random worlds in which X is assigned the sensitive value s. Let $\mathcal{T}^\star_X = \{(\psi, Z_n) \vdash (T^\star, X)\}$ be the set of random worlds which are compatible with $T^\star$ containing X. Let $\mathcal{T}^\star_{(X,s)} = \{(\psi, Z_n) \vdash (T^\star, X) |\ \psi(X) = s\}$ be the set of random worlds compatible with $T^\star$ where X is assigned the sensitive value s. Then,

$$\beta_{(q,s,T^\star)} = \frac{|\mathcal{T}^\star_{(X,s)}|}{|\mathcal{T}^\star_X|}$$

Note that $\mathcal{T}^\star_{(X,s_1)}$ and $\mathcal{T}^\star_{(X,s_2)}$ are disjoint sets of random worlds – in all the worlds in $\mathcal{T}^\star_{(X,s_1)}$, X is assigned the sensitive value s_1 and in all the world in $\mathcal{T}^\star_{(X,s_2)}$, X is assigned the sensitive value s_2. Thus

$$|\mathcal{T}^\star_X| = \sum_{s' \in S} |\mathcal{T}^\star_{(X,s')}|$$

We now proceed to calculate the cardinality of $\mathcal{T}^\star_{(X,s)}$ for each s. First we will compute the number of assignments ψ such that $\psi(X) = s$ and then for each ψ we will compute the number of samples Z_n such that $(\psi, Z_n) \vdash (T^\star, X)$. The number of

assignments ψ compatible with the background knowledge such that $\psi(X) = s$ can be calculated as follows. X is assigned the sensitive value s. Since $X[Q] = q$, out of the remaining $N_q - 1$ individuals having the nonsensitive value q, $N_{(q,s)} - 1$ of them are assigned s. For every other sensitive value s', $N_{(q,s')}$ out of the $N_q - 1$ individuals are assigned s'. For every $q' \neq q$ and every s', some $N_{(q',s')}$ out of the N'_q individuals having the nonsensitive value q' are assigned s'. The number of these assignments is

$$\begin{aligned} & \frac{(N_q - 1)!}{(N_{(q,s)} - 1)! \prod\limits_{s' \neq s} N_{(q,s')}!} \prod_{q' \neq q} \frac{N_{q'}!}{\prod\limits_{s' \in S} N_{(q',s')}!} \\ = \quad & \frac{N_{(q,s)}}{N_q} \prod_{q' \in Q} \frac{N_{q'}!}{\prod\limits_{s' \in S} N_{(q',s')}!} \end{aligned} \tag{2}$$

For each mapping ψ such that $\psi(X) = s$, we count the number of Z_n's such that $(\psi, Z_n) \vdash (T^\star, X)$ as follows. Let $q^\star$ be the generalized value of $q = X[Q]$. X's record will appear as $t^\star_X = (q^\star, s)$ in the table $T^\star$. Apart from $t^\star_X$, $T^\star$ contains $n_{(q^\star,s)} - 1$ other tuples of the form $(q^\star, s)$. Hence, apart from X, Z_n should contain $n_{(q^\star,s)} - 1$ other individuals ω with $\psi(\omega) = s$ and $\omega[Q] = q'$ where q' generalizes to $q^\star$. For all other $(q^{\star\prime}, s')$ such that $q^{\star\prime} \neq q^\star$ or $s' \neq s$, Z_n should contain $n_{(q^{\star\prime},s')}$ individuals ω' where $\psi(\omega') = s'$ and $q^{\star\prime}$ is the generalized value of $\omega[Q]$. The number of Z_n's is given by

$$\begin{aligned} & \binom{N_{(q^\star,s)} - 1}{n_{(q^\star,s)} - 1} \prod_{(q^{\star\prime},s') \in (Q^\star \times S) \setminus \{(q^\star,s)\}} \binom{N_{(q^{\star\prime},s')}}{n_{(q^{\star\prime},s')}} \\ = \quad & \frac{n_{q^\star,s}}{N_{(q^\star,s)}} \prod_{(q^{\star\prime},s') \in Q^\star \times S} \binom{N_{(q^{\star\prime},s')}}{n_{(q^{\star\prime},s')}} \end{aligned} \tag{3}$$

The cardinality of $\mathcal{T}^\star_{(X,s)}$ is therefore the product of Equations 2 and 3 and can be expressed as

$$\begin{aligned} |\mathcal{T}^\star_{(X,s)}| &= \frac{N_{(q,s)}}{N_q} \prod_{q' \in Q} \frac{N_{q'}!}{\prod\limits_{s' \in S} N_{(q',s')}!} \times \frac{n_{q^\star,s}}{N_{(q^\star,s)}} \prod_{(q^{\star\prime},s') \in Q^\star \times S} \binom{N_{(q^{\star\prime},s')}}{n_{(q^{\star\prime},s')}} \\ &= n_{(q^\star,s)} \frac{N_{(q,s)}}{N_{(q^\star,s)}} \times \frac{1}{N_q} \prod_{q' \in Q} \frac{N_{q'}!}{\prod\limits_{s' \in S} N_{(q',s')}!} \times \prod_{(q^{\star\prime},s') \in Q^\star \times S} \binom{N_{(q^{\star\prime},s')}}{n_{(q^{\star\prime},s')}} \\ &= n_{(q^\star,s)} \frac{N_{(q,s)}}{N_{(q^\star,s)}} \times \mathcal{E} \end{aligned}$$

The expression $\mathcal{E}$ is the same for all $s' \in S$. Hence, the expression for the observed belief is

$$\begin{aligned} \beta_{(q,s,T^\star)} &= \frac{|\mathcal{T}^\star_{(X,s)}|}{\sum_{s' \in S} |\mathcal{T}^\star_{(X,s')}|} \\ &= \frac{n_{(q^\star,s)} \frac{N_{(q,s)}}{N_{(q^\star,s)}}}{\sum_{s' \in S} n_{(q^\star,s')} \frac{N_{(q,s')}}{N_{(q^\star,s')}}} \end{aligned}$$

Using the substitutions $f(q,s) = N_{(q,s)}/N$ and $f(q^\star, s) = N_{(q^\star,s)}/N$, we get the required expression.

$$\begin{aligned} \beta_{(q,s,T^\star)} &= \frac{n_{(q^\star,s)} \frac{f(q,s)}{f(q^\star,s)}}{\sum_{s' \in S} n_{(q^\star,s')} \frac{f(q,s')}{f(q^\star,s')}} \\ &= \frac{n_{(q^\star,s)} \frac{f(s|q)}{f(s|q^\star)}}{\sum_{s' \in S} n_{(q^\star,s')} \frac{f(s'|q)}{f(s'|q^\star)}} \end{aligned}$$

Note that in the special case when S and Q are independent, The expression for the observed belief simplifies to

$$\begin{aligned}\beta_{(q,s,T^\star)} &= \frac{n_{(q^\star,s)}\frac{f(s|q)}{f(s|q^\star)}}{\sum_{s'\in S} n_{(q^\star,s')}\frac{f(s'|q)}{f(s'|q^\star)}} \\ &= \frac{n_{(q^\star,s)}\frac{f(s)}{f(s)}}{\sum_{s'\in S} n_{(q^\star,s')}\frac{f(s')}{f(s')}} \\ &= \frac{n_{(q^\star,s)}}{\sum_{s'\in S} n_{(q^\star,s')}}\end{aligned}$$

□

Armed with a way of calculating Alice's belief about Bob's private data after she has seen T^*, let us now examine some principles for building definitions of privacy.

3.2. Privacy Principles

Given the adversary's background knowledge, a published table $T^\star$ might leak private information in two important ways: *positive disclosure* and *negative disclosure*.

Definition 3.1 (Positive disclosure)**.** *Publishing the table $T^\star$ that was derived from T results in a* positive disclosure *if the adversary can correctly identify the value of a sensitive attribute with high probability; i.e., given a $\delta > 0$, there is a positive disclosure if $\beta_{(q,s,T^\star)} > 1 - \delta$ and there exists $t \in T$ such that $t[Q] = q$ and $t[S] = s$.*

Definition 3.2 (Negative disclosure)**.** *Publishing the table $T^\star$ that was derived from T results in a* negative disclosure *if the adversary can correctly eliminate some possible values of the sensitive attribute (with high probability); i.e., given an $\epsilon > 0$, there is a negative disclosure if $\beta_{(q,s,T^\star)} < \epsilon$ and there exists a $t \in T$ such that $t[Q] = q$ but $t[S] \neq s$.*

The homogeneity attack in Section 1.1 where Alice determined that Bob has cancer is an example of a positive disclosure. Similarly, in the example from Section 1.1, even without background knowledge Alice can deduce that Umeko does not have cancer. This is an example of a negative disclosure.

Note that not all positive disclosures are disastrous. If the prior belief was that $\alpha_{(q,s)} > 1 - \delta$, the adversary would not have learned anything new. Similarly, negative disclosures are not always bad: discovering that Bob does not have Ebola might not be very serious because the prior belief of this event was small. Hence, the ideal definition of privacy can be based on the following principle:

Principle 1 (Uninformative Principle)**.** *The published table should provide the adversary with little additional information beyond the background knowledge. In other words, there should not be a large difference between the prior and posterior beliefs.*

The uninformative principle can be instantiated in several ways, for example with the (ρ_1, ρ_2)-privacy breach definition [41].

Definition 3.3 ((ρ_1, ρ_2)-privacy)**.** *Given a table T^* and two constants ρ_1 and ρ_2, we say that a (ρ_1, ρ_2)-privacy breach has occurred when either $\alpha_{(q,s)} < \rho_1 \ \wedge \ \beta_{(q,s,T^\star)} > \rho_2$ or when $\alpha_{(q,s)} > 1 - \rho_1 \ \wedge \ \beta_{(q,s,T^\star)} < 1 - \rho_2$. If a (ρ_1, ρ_2)-privacy breach has not occurred, then table T^* satisfies (ρ_1, ρ_2)-privacy.*

An alternative privacy definition based on the uninformative principle would bound the maximum difference between $\alpha_{(q,s)}$ and $\beta_{(q,s,T^\star)}$ using any of the functions commonly used to measure the difference between probability distributions. Any privacy definition that is based on the uninformative principle, and instantiated either by a (ρ_1, ρ_2)-privacy breach definition or by bounding the difference between $\alpha_{(q,s)}$ and $\beta_{(q,s,T^\star)}$ is a Bayes-optimal privacy definition. The specific choice of definition depends on the application.

Note that any Bayes-optimal privacy definition captures diversity in addition to background knowledge. To see how it captures diversity, suppose that all the tuples whose nonsensitive attribute Q have been generalized to $q^\star$ have the same value s for their sensitive attribute. Then $n_{(q^\star,s')} = 0$ for all $s' \neq s$ and hence the value of the observed belief $\beta_{(q,s,T^\star)}$ becomes 1 in Equation 1. This will be flagged as a breach whenever the prior belief is not close to 1.

3.3. Limitations of the Bayes-Optimal Privacy

For the purposes of our discussion, we are more interested in the properties of Bayes-optimal privacy rather than its exact instantiation. In particular, Bayes-optimal privacy has several drawbacks that make it hard to use in practice.

Insufficient Knowledge. The data publisher is unlikely to know the full distribution f of sensitive and nonsensitive attributes over the general population Ω from which T is a sample.

The Adversary's Knowledge is Unknown. It is also unlikely that the adversary has knowledge of the complete joint distribution between the non-sensitive and sensitive attributes. However, the data publisher does not know how much the adversary knows. For example, in the background knowledge attack in Section 1.1, Alice knew that Japanese have a low incidence of heart disease, but the data publisher did not know that Alice knew this piece of information.

Instance-Level Knowledge. The theoretical definition does not protect against knowledge that cannot be modeled probabilistically. For example, suppose Bob's son tells Alice that Bob does not have diabetes. The theoretical definition of privacy will not be able to protect against such adversaries.

Multiple Adversaries. There will likely be multiple adversaries with different levels of knowledge, each of which is consistent with the full joint distribution. Suppose Bob has a disease that is (a) very likely among people in the age group [30-50], but (b) is very rare for people of that age group who are doctors. An adversary who only knows the interaction of age and illness will think that it is very likely for Bob to have that disease. However, an adversary who also knows that Bob is a doctor is more likely to think that Bob does not have that disease. Thus, although additional knowledge can yield better inferences on average, there are specific instances where it does not. Thus the data publisher must take into account all possible levels of background knowledge.

In the next section, we present a privacy definition that eliminates these drawbacks.

4. ℓ-Diversity: A Practical Privacy Definition

In this section we discuss how to overcome the difficulties outlined at the end of the previous section. We derive the ℓ-diversity principle (Section 4.1), show how to instantiate it with specific definitions of privacy (Section 4.2), outline how to handle multiple sensitive attributes (Section 4.3), and discuss how ℓ-diversity addresses the issues raised in the previous section (Section 4.4).

4.1. The ℓ-Diversity Principle

In this subsection we will derive the principle of ℓ-diversity in two ways. First, we will derive it in an ideal theoretical setting where it can be shown that the adversary's background knowledge will not lead to a privacy breach. Then we will re-derive the ℓ-diversity principle from a more practical starting point and show that even under less-than-ideal circumstances, ℓ-diversity can still defend against background knowledge that is unknown to the data publisher. Although the arguments in this subsection can be made precise, we will keep our discussion at an intuitive level for the sake of clarity.

Let us re-examine the expression for computing the adversary's observed belief (Theorem 3.1):

$$\beta_{(q,s,T^\star)} \quad = \quad \frac{n_{(q^\star,s)} \frac{f(s|q)}{f(s|q^\star)}}{\sum_{s' \in S} n_{(q^\star,s')} \frac{f(s'|q)}{f(s'|q^\star)}} \tag{4}$$

For the moment, let us consider an ideal setting where if two objects have "similar" nonsensitive attributes then their sensitive attributes have similar probabilistic behavior. More formally, given a similarity measure $d(\cdot,\cdot)$ then $\forall \epsilon > 0,\ \exists \delta$ such that if $d(q_1, q_2) < \delta$ then $\max_s |f(s|q_1) - f(s|q_2)| < \epsilon$. This similarity assumption is implicit in all k-Nearest Neighbor classifiers.

Now let us define a *$q^\star$-block* to be the set of tuples in $T^\star$ whose nonsensitive attribute values generalize to $q^\star$. If all tuples in a $q^\star$-block are "similar" based on their nonsensitive attributes, then $f(s|q) \approx f(s|q^\star)$ for those q that appear in the $q^\star$-block, and because of (approximate) cancellations, Equation 4 could be approximated arbitrarily well by Equation 5:

$$L(q, s, T^\star) \quad = \quad \frac{n_{(q^\star,s)}}{\sum_{s' \in S} n_{(q^\star,s')}} \tag{5}$$

Thus given enough data and a good partitioning, background knowledge cancels out and has no effect on the inferences that can be made from the table! The only inferences that can be made are those that depend solely on the $n_{(q^*,s')}$ – the

frequencies of each $s' \in S$ for each q^*-block. Therefore to prevent privacy breaches, we need to ensure for every q^*-block that the ℓ most frequent values of S have roughly the same frequencies. This guarantees that $P(s|q^*) \leq 1/(\ell + \epsilon)$ for some small $\epsilon > 0$ and for all $s \in S$ and ensures that Alice will be uncertain about Bob's true medical condition. This is the essence of ℓ-diversity.

All of those arguments relied on the following three assumptions: tuples with similar non-sensitive attributes values have similar sensitive attributes values, there is a good partitioning of the data, and there is a large amount of data so that many "similar" tuples fall into each partition. Let us re-examine privacy breaches when these assumptions do not hold.

Recall that Theorem 3.1 allows us to calculate the observed belief of the adversary. Consider the case of positive disclosures; i.e., Alice wants to determine that Bob has $t[S] = s$ with very high probability. From Theorem 3.1, this can happen only when:

$$\exists s,\ \forall s' \neq s, \quad n_{(q^\star,s')} \frac{f(s'|q)}{f(s'|q^\star)} \ll n_{(q^\star,s)} \frac{f(s|q)}{f(s|q^\star)} \tag{6}$$

The condition in Equation (6) could occur due to a combination of two factors: (i) a lack of diversity in the sensitive attributes in the $q^\star$-block, and/or (ii) strong background knowledge. Let us discuss these in turn.

Lack of Diversity. Lack of diversity in the sensitive attribute manifests itself as follows:

$$\forall s' \neq s, \quad n_{(q^\star,s')} \ll n_{(q^\star,s)} \tag{7}$$

In this case, almost all tuples have the same value s for the sensitive attribute S, and thus $\beta_{(q,s,T^\star)} \approx 1$. Note that this condition can be easily checked since it only involves counting the values of S in the published table $T^\star$. We can ensure diversity by requiring that *all* the possible values $s' \in domain(S)$ occur in the $q^\star$-block with roughly equal proportions. This, however, is likely to cause significant loss of information: if $domain(S)$ is large then the $q^\star$-blocks will necessarily be large and so the data will be partitioned into a small number of $q^\star$-blocks. Another way to ensure diversity and to guard against Equation 7 is to require that a $q^\star$-block has at least $\ell \geq 2$ different sensitive values such that the ℓ most frequent values (in the $q^\star$-block) have roughly the same frequency. We say that such a $q^\star$-block is *well-represented by ℓ sensitive values*.

Strong Background Knowledge. The other factor that could lead to a positive disclosure (Equation 6) is strong background knowledge. Even though a $q^\star$-block may have ℓ "well-represented" sensitive values, Alice may still be able to use her background knowledge to eliminate sensitive values when the following is true:

$$\exists s', \quad \frac{f(s'|q)}{f(s'|q^\star)} \approx 0 \tag{8}$$

This equation states that Bob with quasi-identifier $t[Q] = q$ is much less likely to have sensitive value s' than any other individual in the $q^\star$-block. For example, Alice may know that Bob never travels, and thus he is extremely unlikely to have Ebola. It is not possible for a data publisher to reveal some information about the data while still guarding against attacks employing arbitrary amounts of background knowledge (since the revealed information may be precisely what the adversary needs to recreate the entire table). However, the data publisher can still guard against many attacks even without having access to Alice's background knowledge. In our model, Alice might know the distribution $f(q, s)$ over the sensitive and non-sensitive attributes, in addition to the conditional distribution $f(s|q)$. The most damaging type of such information has the form $f(s|q) \approx 0$, e.g., "men do not have breast cancer", or the form of Equation 8, e.g., "Japanese have a very low incidence of heart disease". Note that *a priori* information of the form $f(s|q) = 1$ is not as harmful since this positive disclosure is independent of the published table $T^\star$. Alice can also eliminate sensitive values with instance-level knowledge such as "Bob does not have diabetes".

In spite of such background knowledge, if there are ℓ "well represented" sensitive values in a $q^\star$-block, then Alice needs $\ell - 1$ damaging pieces of background knowledge to eliminate $\ell - 1$ possible sensitive values and infer a positive disclosure! Thus, by setting the parameter ℓ, the data publisher can determine how much protection is provided against background knowledge — even if this background knowledge is unknown to the publisher.

Note that Alice may know ℓ pieces of instance-level background knowledge of the form "individual X_i does not have disease Y" (for $i = 1 \ldots \ell$), where each X_i is a different individual. However, we have been talking only about eliminating sensitive values for a single individual. It has been shown [55] that for a specific individual Bob, the worst case disclosure occurs when $X_i =$ Bob in all the ℓ pieces of information Alice possesses.

Moreover, when inferring information about Bob, knowing the exact sensitive values of some other individuals in the table is less damaging than statements of the form "Bob does not have cancer". This is because knowing the sensitive value

	Non-Sensitive			Sensitive
	Zip Code	Age	Nationality	Condition
1	1305*	≤ 40	$*$	Heart Disease
4	1305*	≤ 40	$*$	Viral Infection
9	1305*	≤ 40	$*$	Cancer
10	1305*	≤ 40	$*$	Cancer
5	1485*	> 40	$*$	Cancer
6	1485*	> 40	$*$	Heart Disease
7	1485*	> 40	$*$	Viral Infection
8	1485*	> 40	$*$	Viral Infection
2	1306*	≤ 40	$*$	Heart Disease
3	1306*	≤ 40	$*$	Viral Infection
11	1306*	≤ 40	$*$	Cancer
12	1306*	≤ 40	$*$	Cancer

Figure 4. 3-Diverse Inpatient Microdata

for some other individual only eliminates from consideration one tuple that may have corresponded to Bob while the latter statement eliminates *at least* one tuple.

Putting these two arguments together, we arrive at the following principle.

Principle 2 (ℓ-Diversity Principle)**.** *A $q^\star$-block is ℓ-diverse if contains at least ℓ "well-represented" values for the sensitive attribute S. A table is ℓ-diverse if every $q^\star$-block is ℓ-diverse.*

Returning to our example, consider the inpatient records shown in Figure 1. We present a 3-diverse version of the table in Figure 4. Comparing it with the 4-anonymous table in Figure 2 we see that the attacks against the 4-anonymous table are prevented by the 3-diverse table. For example, Alice cannot infer from the 3-diverse table that Bob (a 31 year old American from zip code 13053) has cancer. Even though Umeko (a 21 year old Japanese from zip code 13068) is extremely unlikely to have heart disease, Alice is still unsure whether Umeko has a viral infection or cancer.

The ℓ-diversity principle advocates ensuring ℓ "well represented" values for the sensitive attribute in every $q^\star$-block, but does not clearly state what "well represented" means. Note that we called it a "principle" instead of a definition — we will use it to give two concrete instantiations of the ℓ-diversity principle and discuss their relative trade-offs.

4.2. ℓ-Diversity: Instantiations

In this section we will give two instantiations of the ℓ-diversity principle: entropy ℓ-diversity and recursive ℓ-diversity. After presenting the basic definitions, we'll extend them to cases where some positive disclosure is allowed.

The first instantiation of the ℓ-diversity principle, and the simplest one to describe, uses the information-theoretic notion of entropy:

Definition 4.1 (Entropy ℓ-Diversity [60])**.** *A table is* Entropy ℓ-Diverse *if for every $q^\star$-block*

$$-\sum_{s \in S} p_{(q^\star,s)} \log(p_{(q^\star,s')}) \geq \log(\ell)$$

where $p_{(q^\star,s)} = \frac{n_{(q^\star,s)}}{\sum_{s' \in S} n_{(q^\star,s')}}$ is the fraction of tuples in the $q^\star$-block with sensitive attribute value equal to s.

As a consequence of this condition, every $q^\star$-block has at least ℓ distinct values for the sensitive attribute. Using this definition, Figure 4 is actually 2.8-diverse.

Entropy ℓ-diversity was first proposed by Ohrn et al. [60] as a way of defending against the homogeneity problem (without considering the role of background knowledge). Note that entropy ℓ-diversity captures the notion of well-represented groups due to the fact that entropy increases as frequencies become more uniform. We can also capture the role of background knowledge more explicitly with an alternate definition.

Let $s_1, \ldots, s_m$ be the possible values of the sensitive attribute S in a $q^\star$-block. Assume that we sort the counts $n_{(q^\star,s_1)}$, $\ldots$, $n_{(q^\star,s_m)}$ in descending order and name the elements of the resulting sequence $r_1, \ldots, r_m$. One way to think about ℓ-diversity is the following: the adversary needs to eliminate at least $\ell - 1$ possible values of S in order to infer a positive disclosure. This means that, for example, in a 2-diverse table, none of the sensitive values should appear too frequently. We say that a $q^\star$-block is $(c, 2)$-diverse if $r_1 < c(r_2 + \cdots + r_m)$ for some user-specified constant c. For $\ell > 2$, we say that a $q^\star$-block satisfies *recursive* (c, ℓ)*-diversity* if we can eliminate one possible sensitive value in the $q^\star$-block and still have a $(c, \ell-1)$-diverse block. This recursive definition can be succinctly stated as follows:

Definition 4.2 (Recursive (c, ℓ)-Diversity)**.** *In a given $q^\star$-block, let r_i denote the number of times the i^{th} most frequent sensitive value appears in that $q^\star$-block. Given a constant c, the $q^\star$-block satisfies* recursive (c, ℓ)-diversity *if $r_1 < c(r_\ell + r_{\ell+1} + \cdots + r_m)$. A table $T^\star$ satisfies recursive (c, ℓ)-diversity if every $q^\star$-block satisfies recursive ℓ-diversity. We say that* 1*-diversity is always satisfied.*

Now, both entropy and recursive ℓ-diversity may be too restrictive. To see why, let us first look at entropy ℓ-diversity. Since $-x \log(x)$ is a concave function, it can be shown that if we split a $q^\star$-block into two sub-blocks $q_a^\star$ and $q_b^\star$ then $\text{entropy}(q^\star) \geq \min(\text{entropy}(q_a^\star), \text{entropy}(q_b^\star))$. This implies that in order for entropy ℓ-diversity to be possible, the entropy of the entire table must be at least $\log(\ell)$. This might not be the case, especially if one value of the sensitive attribute is very common – for example, if 90% of the patients have "heart problems" as the value for the "Medical Condition" attribute.

This is also a problem with recursive ℓ-diversity. It is easy to see that if 90% of the patients have "heart problems" as the value for the "Medical Condition" attribute then there will be at least one q^*-block where "heart problems" will have frequency of at least 90%. Therefore if we choose $c < 9$ in Definition 4.2, no generalization of the base table will satisfy recursive (c, ℓ)-diversity.

One the other hand, some positive disclosures may be acceptable. For example, a clinic might be allowed to disclose that a patient has a "heart problem" because it is well known that most patients who visit the clinic have heart problems. It may also be allowed to disclose that "Medical Condition" = "Healthy" if this is not considered an invasion of privacy.

At this point one may be tempted to remove tuples with nonsensitive "Medical Condition" values, publish them unaltered, and then create an ℓ-diverse version of the remaining dataset. In some cases this is acceptable. However, there are three important issues why the above suggestion may not be acceptable: the anonymity of the unaltered tuples, the privacy of the remaining tuples, and the utility of the resulting published data.

First, publishing unaltered tuples gives an adversary the ability to link them to external data and identify the corresponding individuals. This may be considered a privacy breach [20], since it is reasonable for individuals to object to being identified as respondents in a survey. To avoid this one could publish a k-anonymous version of tuples with nonsensitive "Medical Condition" values and a ℓ-diverse version of the rest of the table.

Second, separating individuals with nonsensitive medical conditions from the rest can impact the individuals with *sensitive* medical conditions. As an extreme case, suppose "Medical Condition" can only take two values: "Healthy" and "Sick". There is no way to achieve 2-diversity on the table of patients that are sick; if Alice knows Bob is in the table and Bob is not listed as a healthy patient, he must then be sick. More generally, separating records with sensitive values from records with nonsensitive values reduces the possible choices for the security parameter ℓ.

A third issue with partitioning the data into two tables is related to the utility of the data for a researcher. Since each of the tables is smaller than the whole dataset, to satisfy k-anonymity and ℓ-diversity the tables might have to be generalized more than if a single table had been anonymized. For instance, consider a table reporting the "Gender" and "Medical Condition" of 2,000 individuals, where the attribute "Medical Condition" can take three values: "Healthy", "Cancer", and "Hepatitis". In this table there are 1,000 males and 1,000 females. 700 of the 1,000 males are "Healthy" and the other 300 have "Hepatitis". 700 of the 1,000 females are "Healthy" while the other 300 have "Cancer". If the disclosure of "Medical Condition" = "Healthy" is not considered an invasion of privacy, then this table satisfies 2-diversity (and thus requires no further generalizations). In contrast, if we were to publish the "Healthy" patients separately, we would need to suppress the gender information of the unhealthy individuals in order to achieve 2-diversity on the table containing the unhealthy patients.

Additionally, if the data is separated then the two resulting tables are likely to have different schemas. For example, one table may be generalized so that "Age" appears as an interval of length 5 (i.e. [30-34]) and only the first 4 digits of "Zip Code" are given, while the second table may give the full "Zip Code" but may generalize "Age" to intervals of length 10. Learning from such data is not as straightforward as learning from a single table.

Thus an alternate approach is needed to handle the case when some of the values in the domain of the sensitive attribute need not be kept private. To capture this notion that some positive disclosure is acceptable, let Y be the set of those sensitive values for which positive disclosure is allowed. We call Y a *don't-care* set. Note that we are not worried about those values

being too frequent. Let s_y be the most frequent sensitive value in the $q^\star$-block that is *not* in Y and let r_y be the associated frequency. Then the $q^\star$-block satisfies ℓ-diversity if we can eliminate the $\ell - 2$ most frequent values of S *not including* r_y without making s_y too frequent in the resulting set. This is the same as saying that after we remove the sensitive values with counts $r_1, \ldots, r_{y-1}$, then the result is $(\ell - y + 1)$-diverse. This brings us to the following definition.

Definition 4.3. (Positive Disclosure-Recursive (c, ℓ)-Diversity). *Let $Y \subset S$ be a don't-care set. In a given $q^\star$-block, let the most frequent sensitive value not in Y be the y^{th} most frequent sensitive value. Let r_i denote the frequency of the i^{th} most frequent sensitive value in the $q^\star$-block. Such a $q^\star$-block satisfies* pd-recursive (c, ℓ)-diversity *if one of the following hold:*

- $y \leq \ell - 1$ *and* $r_y < c \sum_{j=\ell}^{m} r_j$
- $y > \ell - 1$ *and* $r_y < c \sum_{j=\ell-1}^{y-1} r_j + c \sum_{j=y+1}^{m} r_j$

We denote the summations on the right hand side of the both conditions by $tail_{q^\star}(s_y)$.

Now, note that if $r_y = 0$ then the $q^\star$-block only has sensitive values that can be disclosed and so both conditions in Definition 4.3 are trivially satisfied. Second, note that if $c > 1$ then the second condition clearly reduces to just the condition $y > \ell - 1$ because $r_y \leq r_{\ell-1}$. The second condition states that even though the $\ell - 1$ most frequent values can be disclosed, we still do not want r_y to be too frequent if $\ell - 2$ of them have been eliminated (i.e., we want the result to be 2-diverse).

To see this definition in action, suppose there are two values for "Medical Condition", *healthy* and *not healthy*. If *healthy* is a don't-care value, then $(c, 2)$-diversity states that the number of sick patients in a q^*-block is less than c times the number of healthy patients or, equivalently, at most $\frac{c}{c+1}$ patients in a q^*-block are sick. Thus if $c = 0.03$ then at most 3% of the patients in any q^*-block are not healthy, and if $c = 1$ then at most half the patients in any q^*-block are not healthy.

Entropy ℓ-diversity can also be extended to handle don't-care sets. The description of entropy ℓ-diversity with don't-care sets is a bit more involved, so before we present it, we shall briefly touch upon the subject of negative disclosure.

Until now we have treated negative disclosure as relatively unimportant compared to positive disclosure. However, negative disclosure may also be important. If W is the set of values for the sensitive attribute for which negative disclosure is not allowed then, given a user-specified constant $c_2 < 100$, we require that each $s \in W$ appear in at least c_2-percent of the tuples in every $q^\star$-block, resulting in the following definition. This is incorporated into ℓ-diversity definitions in a straightforward way:

Definition 4.4. (Negative/Positive Disclosure-Recursive (c_1, c_2, ℓ)-Diversity). *Let W be the set of sensitive values for which negative disclosure is not allowed. A table satisfies* npd-recursive (c_1, c_2, ℓ)-diversity *if it satisfies pd-recursive (c_1, ℓ)-diversity and if every $s \in W$ occurs in at least c_2 percent of the tuples in every $q^\star$-block.*

We now conclude this subsection with a definition of entropy ℓ-diversity that uses don't-care sets. The extension of entropy ℓ-diversity is more complicated than for recursive ℓ-diversity, but the motivation is similar. Let S be a sensitive attribute. Suppose we have a q^*-block q_A where the values of S are $s_1, s_2, \ldots, s_n$ with corresponding counts $p_1, \ldots, p_n$ (note that unlike before, we don't require the counts to be sorted; thus p_i is shorthand for $n_{(q_A, s_i)}$). Furthermore, suppose s_1 belongs to the don't-care set so that we can safely disclose the value of S when it equals s_1. If in this hypothetical q^*-block, 90% of the tuples have sensitive value s_1, then this block has a low entropy. Now consider a q^*-block q_B with sensitive values $s_1, s_2, \ldots, s_n$ with counts $p'_1, p_2, p_3, \ldots, p_n$ (where $p'_1 > p_1$). The block q_B is just like q_A except that there are more tuples with the don't-care value s_1.

Intuitively, since s_1 is a don't-care value, q_B cannot pose more of a disclosure risk that q_A. Thus if we were free to adjust the value p_1, we should expect that disclosure risk does not decrease when we decrease p_1 and disclosure risk does not increase when we increase p_1. Treating p_1 as a variable, let's *lower* it from its initial setting in q_A to the unique value p^* that would maximize the entropy of the q^*-block. The original disclosure risk of q_A cannot be any higher than the disclosure risk at the optimum value p^*. We will compute the entropy at this optimum value p^* and set the disclosure risk of q_A to be this value. In the more general case (with more than one don't-care value), we determine what is the maximum entropy we would get if we lowered the counts corresponding to don't-care values from their initial values. We call this maximum entropy value the *adjusted entropy* and it will serve as the disclosure risk of the q^*-block: if the adjusted entropy is larger than $\log \ell$ then the block is considered ℓ-diverse.

Before we formalize this, we should note that this type of argument will also yield our original definition for recursive ℓ-diversity in the presence of don't-care sets. One can easily check that if p'' is the count of the most frequent sensitive

value (not in the don't care set) and $\phi_1, \ldots, \phi_r$ are the counts of don't-care values that appear more frequently, the recursive ℓ-diversity procedure for don't-care sets lowers the values $\phi_1, \ldots, \phi_r$ to set them equal to p'', and then checks if the resulting block satisfies ordinary recursive ℓ-diversity.

To formalize the notion of adjusted entropy, we need the following notation. For nonnegative values $x_1, \ldots, x_n$ such that $\sum x_i = 1$, denote the entropy as:

$$H(x_1, \ldots, x_n) = -\sum_{i=1}^{m} x_i \log x_i$$

with the understanding that $0 \log 0 = 0$. For arbitrary nonnegative numbers $x_1, \ldots, x_n$, denote the *normalized entropy* as:

$$\hat{H}(x_1, \ldots, x_n) = -\sum_{i=1}^{n} \frac{x_i}{\sum_{j=1}^{n} x_j} \log \left(\frac{x_i}{\sum_{j=1}^{n} x_j} \right) \tag{9}$$

First, we define adjusted entropy, and then show how to compute it.

Definition 4.5 (Adjusted Entropy)**.** *Let S be a sensitive attribute with don't-care values $y_1, \ldots, y_r$ and sensitive values $s_1, \ldots, s_m$. Let q_A be a q^*-block where the don't-care values y_i have counts ϕ_i and the sensitive values s_j have counts p_j. The* adjusted entropy *of q_A is defined as:*

$$\sup_{0 \le x_i \le \phi_i;\ i=1,\ldots,r} \hat{H}(x_1, \ldots, x_r, p_1, \ldots, p_m) \tag{10}$$

The maximizing values of the x_i in Definition 4.5 are closely related to the function

$$M(c_1, \ldots, c_k) = \frac{\sum_{i=1}^{k} c_i \log c_i}{\sum_{i=1}^{k} c_i}$$

which we call the *log-entropic mean* of $c_1, \ldots, c_k$ (because it is the weighted average of their logarithms).[3] We show that there exists a unique vector $(c_1, c_2, \ldots, c_r)$ that maximizes Equation 10 and we can characterize it with the following theorem:

Theorem 4.1. *There is a unique vector $(c_1, c_2, \ldots, c_r)$ such that the assignment $x_i = c_i$ maximizes Equation 10. Furthermore, let $\theta = \max(\{\phi_i \mid c_i = \phi_i\} \cup \{0\})$. If $\phi_j \le \theta$ then $c_j = \phi_j$. If $\phi_j > \theta$ then $\log c_j$ is the log-entropic mean of the set $\{p_1, \ldots, p_m\} \cup \{\phi_i \mid \phi_i = c_i\}$, and θ is the minimum value for which this condition can be satisfied.*

The proof of this theorem is rather technical and can be found in Appendix A. This theorem tells us that some coordinates will achieve their upper bound ϕ_i (i.e., they will not be lowered from their initial values). We call these the *fixed* coordinates. The rest of the coordinates, called the *changeable* coordinates, will be adjusted down until their logarithms equal the log-entropic mean of the fixed coordinates and the counts of the sensitive values (in particular, it means that if c_j is the value of an unchangeable coordinate, then $\log \phi_j$ must be larger than that log-entropic mean). The theorem also tells us that there is a cutoff value θ such that all coordinates with with upper bound $> \theta$ will be changeable and the rest will be fixed. Finally, the theorem also tells us that we should choose the minimum cutoff value for which this is possible.

The computation of adjusted entropy is shown in Algorithm 1. We illustrate the algorithm with a sample run-through. Suppose there are four don't-care values y_1, y_2, y_3, and y_4 with counts 11, 10, 3, and 2, respectively; and suppose there are two sensitive values s_1 and s_2 with counts 3 and 4, respectively. Initially we compute the log-entropic mean of s_1 and s_2, which is 1.263. Now, y_4 has the smallest count among don't-care values and $\log y_4 = 0.693$ which is less than the log-entropic mean. We conclude that y_4 is a fixed value, and we compute the log-entropic mean of $\{y_4, s_1, s_2\}$, which is 1.136. Now, y_3 has the next smallest count among don't-care values. The value $\log y_3$ is 1.099, which is less than the new log-entropic mean. Thus y_3 is also fixed and we compute the log-entropic mean of $\{y_4, y_3, s_1, s_2\}$ which is 1.127. The next value we consider is y_2. Now $\log y_2 = 2.30$ which is greater than the log-entropic mean. Thus y_2 and y_1 are the changeable

[3]Note that the log-entropic mean is the logarithm of a weighted geometric mean of the c_i, which itself belongs to a general class of means called the *entropic means* [17].

Algorithm 1 : AdjustedEntropy($\phi_1, \ldots, \phi_r, p_1, \ldots, p_m$)

Require: $\phi_i \geq 0, p_j \geq 0$

for all $i = 1, \ldots, r$ **do**
  $x_i \leftarrow \phi_i$
end for
fixed $\leftarrow \{p_1, \ldots, p_m\}$
changeable $\leftarrow \{x_1, \ldots, x_r\}$
$m \leftarrow M(\text{fixed})$
while $\log(\min(\text{changeable})) < m$ **do**
  $i = \text{argmin}_{j:x_j \in \text{changeable}} x_j$
  fixed = fixed $\cup \{x_i\}$
  changeable = changeable $\setminus \{x_i\}$
  $m \leftarrow M(\text{fixed})$
end while
for all $x_i \in$ changeable **do**
  $x_i \leftarrow e^m$
end for
return $\hat{H}(x_1, \ldots, x_r, p_1, \ldots, p_m)$

values and the cutoff θ described by Theorem 4.1 must be 3 (the value of y_3). Thus the adjusted entropy should be the normalized entropy of $\{e^{1.127}, e^{1.127}, y_3, y_4, s_1, s_2\}$.

Clearly the definition of adjusted entropy is consistent with entropy ℓ-diversity when there are no don't-care values. Thus to verify correctness of the algorithm, we just need to prove Theorem 4.1. The interested reader may find the proof in Appendix A.

4.3. Multiple Sensitive Attributes

Multiple sensitive attributes present some additional challenges. Suppose S and V are two sensitive attributes, and consider the $q^\star$-block with the following tuples: $\{(q^\star, s_1, v_1), (q^\star, s_1, v_2), (q^\star, s_2, v_3), (q^\star, s_3, v_3)\}$. This $q^\star$-block is 3-diverse (actually recursive (2,3)-diverse) with respect to S (ignoring V) and 3-diverse with respect to V (ignoring S). However, if we know that Bob is in this block and his value for S is not s_1 then his value for attribute V cannot be v_1 or v_2, and therefore must be v_3. One piece of information destroyed his privacy. Thus we see that *a q^*-block that is ℓ-diverse in each sensitive attribute separately may still violate the principle of ℓ-diversity.*

Intuitively, the problem occurred because within the q^*-block, V was not well-represented for each value of S. Had we treated S as part of the quasi-identifier when checking for diversity in V (and vice versa), we would have ensured that the ℓ-diversity principle held for the entire table. Formally,

Definition 4.6 (Multi-Attribute ℓ-Diversity)**.** *Let T be a table with nonsensitive attributes $Q_1, \ldots, Q_{m_1}$ and sensitive attributes $S_1, \ldots, S_{m_2}$. We say that T is ℓ-diverse if for all $i = 1 \ldots m_2$, the table T is ℓ-diverse when S_i is treated as the sole sensitive attribute and $\{Q_1, \ldots, Q_{m_1}, S_1, \ldots, S_{i-1}, S_{i+1}, \ldots, S_{m_2}\}$ is treated as the quasi-identifier.*

As the number of sensitive attributes grows, it is not hard to see that we will necessarily need larger and larger q^*-blocks to ensure diversity. This problem may be ameliorated through tuple suppression, generalization on the sensitive attributes, and publishing marginals (rather than the full table) containing different sensitive attributes. This is a subject for future work.

4.4. Discussion

Recall that we started our journey into Section 4 motivated by the weaknesses of Bayes-optimal privacy. Let us now revisit these issues one by one.

- ℓ-Diversity no longer requires knowledge of the full distribution of the sensitive and nonsensitive attributes.
- ℓ-Diversity does not even require the data publisher to have as much information as the adversary. The parameter ℓ protects against more knowledgeable adversaries; the larger the value of ℓ, the more information is needed to rule out possible values of the sensitive attribute.

- Instance-level knowledge (Bob's son tells Alice that Bob does not have diabetes) is automatically covered. It is treated as just another way of ruling out possible values of the sensitive attribute.
- Different adversaries can have different background knowledge leading to different inferences. ℓ-Diversity simultaneously protects against all of them without the need for checking which inferences can be made with which levels of background knowledge.

Overall, we believe that ℓ-diversity is practical, easy to understand, and addresses the shortcomings of k-anonymity with respect to the background knowledge and homogeneity attacks. Let us now see whether we can give efficient algorithms to implement ℓ-diversity. We will see that, unlike Bayes-optimal privacy, ℓ-diversity possesses a property called *monotonicity*. We define this concept in Section 5, and we show how this property can be used to efficiently generate ℓ-diverse tables.

5. Implementing Privacy-Preserving Data Publishing

In this section we discuss how to build algorithms for privacy-preserving data publishing using domain generalization. Let us first review the search space for privacy-preserving data publishing using domain generalization [12, 51]. For ease of explanation, we will combine all the nonsensitive attributes into a single multi-dimensional attribute Q. For attribute Q, there is a user-defined generalization lattice. Formally, we define a generalization lattice to be a set of domains partially ordered by a generalization relation $\prec_G$ (as described in Section 2). The bottom element of this lattice is $\mathrm{domain}(Q)$ and the top element is the domain where each dimension of Q is generalized to a single value. Given a base table T, each domain $D^\star_Q$ in the lattice defines an anonymized table $T^\star$ which is constructed by replacing each tuple $t \in T$ by the tuple $t^\star$, such that the value $t^\star[Q] \in D^\star_Q$ is the generalization of the value $t[Q] \in \mathrm{domain}(Q)$. An algorithm for data publishing should find a point on the lattice such that the corresponding generalized table $T^\star$ preserves privacy and retains as much utility as possible. In the literature, the utility of a generalized table is usually defined as a distance metric on the lattice – the closer the lattice point is to the bottom, the larger the utility of the corresponding table $T^\star$. Hence, finding a a suitable anonymized table $T^\star$ is essentially a lattice search problem. There has been work on search strategies for k-anonymous tables that explore the lattice top-down [12] or bottom-up [51].

In general, searching the entire lattice is computationally intractable. However, lattice searches can be made efficient if there is a stopping condition of the form: if $T^\star$ preserves privacy then every generalization of $T^\star$ also preserves privacy [51, 63]. This is called the *monotonicity property*, and it has been used extensively in frequent itemset mining algorithms [8]. k-Anonymity satisfies the monotonicity property, and it is this property which guarantees the correctness of all efficient algorithms [12, 51]. Thus, if we show that ℓ-diversity also possesses the monotonicity property, then we can re-use these efficient lattice search algorithms to find the ℓ-diverse table with optimal utility. The same cannot be said of Bayes-optimal privacy; the following theorem gives a computational reason why Bayes-optimal privacy does not lend itself to efficient algorithmic implementations.

Theorem 5.1. *Bayes-optimal privacy does not satisfy the monotonicity property.*

Proof. We shall prove this theorem for the $\rho_1 - \rho_2$ version of the Bayes-optimal privacy definition (see Definition 3.3 and [41]); the proof can easily be extended to other instantiations. We set $\rho_1 = 0.31$ and $\rho_2 = 0.58$ and we will create an example where the prior belief $a_{(q,s)} < \rho_1$ but the observed belief is $\beta_{(q,s,T^\star)} > \rho_2$.

First consider Figure 5 which shows a base table T with two values for Q and two values for S.

	q_1	q_2
s_1	$f(q_1, s_1) = .15$ $n_{(q_1,s_1)} = 1$	$f(q_2, s_1) = .25$ $n_{(q_2,s_1)} = 35$
s_2	$f(q_1, s_2) = .35$ $n_{(q_1,s_2)} = 1$	$f(q_2, s_2) = .25$ $n_{(q_2,s_2)} = 15$

Figure 5. Table T

Based on this information, we can compute the prior and observed beliefs for table T.

- $\alpha_{(q_1,s_1)} = .3, \beta_{(q_1,s_1,T)} = .5$

- $\alpha_{(q_1,s_2)} = .7, \beta_{(q_1,s_2,T)} = .5$
- $\alpha_{(q_2,s_1)} = .5, \beta_{(q_2,s_1,T)} = .7$
- $\alpha_{(q_2,s_2)} = .5, \beta_{(q_2,s_2,T)} = .3$

Clearly, publishing T does not breach privacy. However, suppose we generalized T by generalizing both q_1 and q_2 to $q^\star$, as in Figure 6:

	$q^\star$
s_1	$f(q^\star, s_1) = .4$ $n_{(q^\star,s_1)} = 36$
s_2	$f(q^\star, s_2) = .6$ $n_{(q^\star,s_2)} = 16$

Figure 6. Table $T^\star$

If Bob has nonsensitive value q_1, then as before, $\alpha_{(q_1,s_1)} = .3 < \rho_1$. However,

$$\beta_{(q_1,s_1,T^\star)} = \frac{36\frac{.15}{.4}}{36\frac{.15}{.4} + 16\frac{.35}{.6}} > \frac{13.5}{13.5 + 9.34} > .59 > \rho_2$$

Thus while publishing T would not cause a privacy breach, publishing $T^\star$ would. This counterexample proves that Bayes-optimal privacy is not monotonic. □

This seemingly counterintuitive result has a simple explanation. Note that there are many more tuples t with $t[Q] = q_2$ than there are with $t[Q] = q_1$. This causes the probabilistic behavior of the $q^\star$-block in $T^\star$ to be heavily influenced by the tuples with $t[Q] = s_2$ and so it "pulls" the value of $\beta_{(q_1,s_1,T^\star)} = \beta_{(q_2,s_1,T^\star)}$ closer to $\beta_{(q_2,s_1,T)}$ (this can be verified with Equation 1 for observed belief). Since the prior belief $\alpha_{(q_1,s_1)}$ doesn't change and since $\alpha_{(q_1,s_1)}$ and $\alpha_{(q_2,s_1)}$ are very different, we get a privacy breach from publishing $T^\star$ but not from publishing T.

Theorem 5.2 (Monotonicity of entropy ℓ-diversity)**.** *Entropy ℓ-diversity satisfies the monotonicity property: if a table $T^\star$ satisfies entropy ℓ-diversity, then any generalization $T^{\star\star}$ of $T^\star$ also satisfies entropy ℓ-diversity.*

Theorem 5.2 follows from the fact that entropy is a concave function. Thus if the $q^\star$-blocks $q_1^\star, \ldots, q_d^\star$ from table $T^\star$ are merged to form the $q^\star$-block $q^{\star\star}$ of table $T^{\star\star}$, then the $\text{entropy}(q^{\star\star}) \geq \min_i(\text{entropy}(q_i^\star))$.

Theorem 5.3 (Monotonicity of npd recursive ℓ-diversity)**.** *The npd recursive (c_1, c_2, ℓ)-diversity criterion satisfies the monotonicity property: if a table $T^\star$ satisfies npd recursive (c_1, c_2, ℓ)-diversity, then any generalization $T^{\star\star}$ of $T^\star$ also satisfies npd recursive (c_1, c_2, ℓ)-diversity.*

Proof. We shall prove this for the case where T^{**} is derived from T^* by merging two $q^\star$-blocks; the general case follows by induction. Let $q_a^\star$ and $q_b^\star$ be the $q^\star$-blocks of $T^\star$ that are merged to form the $q^\star$-block $q^{\star\star}$ of table $T^{\star\star}$. The frequencies of the sensitive values in $q^{\star\star}$ is the sum of the corresponding frequencies in $q_a^\star$ and $q_b^\star$.

First, let us consider negative disclosures. If every sensitive value $s \in W$ occurs in at least c_2 percent of the tuples in $q_a^\star$ and $q_b^\star$, then surely s should also occur in at least a c_2 percent of the tuples in the $q^{\star\star}$.

Next let us consider positive disclosures. Let Y be the set of sensitive values for which positive disclosure is allowed. Let s_y be the most frequent sensitive value in $q^{\star\star}$ that does not appear in Y. Let s_{y_a} and s_{y_b} be the most frequent sensitive values in $q_a^\star$ and $q_b^\star$, respectively, which are not in Y. Clearly if r_y, r_{y_a} and r_{y_b} are the respective counts, then

$$r_y \leq r_{y_a} + r_{y_b}$$

We also know that the $q^\star$-blocks $q_a^\star$ and $q_b^\star$-block are (c_1, ℓ)-diverse (by hypothesis). Hence

$$r_{y_a} \leq c_1 \,\text{tail}_{q_a^\star}(s_{y_a})$$

$$r_{y_b} \leq c_1 \,\text{tail}_{q_b^\star}(s_{y_b})$$

We are done if we prove that $r_y \leq c_1 \operatorname{tail}_{q^\star}(s_y)$. Since s_{y_a} is at least as frequent as s_y in $q_a^\star$ (and similarly for s_{y_b}) then by the definition of $\operatorname{tail}_{q^\star}$, we have

$$\begin{aligned}\operatorname{tail}_{q_a^\star}(s_y) &\geq \operatorname{tail}_{q_a^\star}(s_{y_a})\\ \operatorname{tail}_{q_b^\star}(s_y) &\geq \operatorname{tail}_{q_b^\star}(s_{y_b})\\ \operatorname{tail}_{q^{\star\star}}(s_y) &= \operatorname{tail}_{q_a^\star}(s_y) + \operatorname{tail}_{q_b^\star}(s_y)\end{aligned}$$

Hence

$$\begin{aligned} r_y &\leq r_{y_a} + r_{y_b}\\ &\leq c_1(\operatorname{tail}_{q_a^\star}(s_{y_a}) + \operatorname{tail}_{q_b^\star}(s_{y_b}))\\ &\leq c_1(\operatorname{tail}_{q_a^\star}(s_y) + \operatorname{tail}_{q_b^\star}(s_y))\\ &= c_1 \operatorname{tail}_{q^{\star\star}}(s_y)\end{aligned}$$

and so the $q^\star$-block $q^{\star\star}$ is npd (c_1, c_2, ℓ)-diverse. □

We can also show that entropy ℓ-diversity with don't-care sets satisfies the monotonicity property and is therefore amenable to efficient algorithms. We will first need the following two results which will let us conclude that $\hat{H}(\vec{x}+\vec{y}) \geq \min(\hat{H}(\vec{x}), \hat{H}(\vec{y}))$.

Lemma 5.1. *Let $a_1, \ldots, a_n$ be nonnegative numbers that add up to 1. Let $b_1, \ldots, b_n$ be nonnegative numbers that add up to 1. Then for any $t \in [0, 1]$,*

$$\begin{aligned}\hat{H}(ta_1 + (1-t)b_1, \ldots, ta_n + (1-t)b_n) &= -\sum_{i=1}^{n}[ta_i + (1-t)b_i]\log[ta_i + (1-t)b_i]\\ &\geq -t\sum_{i=1}^{n} -a_i \log a_i \quad - (1-t)\sum_{i=1}^{n} b_i \log b_i\\ &= t\hat{H}(a_1, \ldots, a_n) + (1-t)\hat{H}(b_1, \ldots, b_n)\\ &\geq \min\left(\hat{H}(a_1, \ldots, a_n), \hat{H}(b_1, \ldots, b_n)\right)\end{aligned}$$

with the understanding that $0 \log 0 = 0$.

Proof. This follows immediately from the fact that $-x \log x$ is concave. □

Corollary 5.1. *Let $a_1, \ldots, a_n$ be nonnegative numbers (at least one of which is nonzero) and let $b_1, \ldots, b_n$ be nonnegative numbers (at least one of which is nonzero). Then*

$$\hat{H}(a_1 + b_1, a_2 + b_2, \ldots, a_n + b_n) \geq \min\left(\hat{H}(a_1, \ldots, a_n), \hat{H}(b_1, \ldots, b_n)\right)$$

Proof. Let $A = \sum_{i=1}^{n} a_i$ and $B = \sum_{i=1}^{n} b_i$. By definition, $\hat{H}(a_1, \ldots, a_n) = H(a_1/A, \ldots, a_n/A)$, $\hat{H}(b_1, \ldots, b_n) = \hat{H}(b_1/B, \ldots, b_n/B)$ and $\hat{H}(a_1 + b_1, \ldots, a_n + b_n) = \hat{H}((a_1 + b_1)/(A + B), \ldots, (a_n + b_n)/(A + B))$. Furthermore, let $t = A/(A + B)$. Then $(a_i + b_i)/(A + B) = t(a_i/A) + (1 - t)(b_i/B)$. Applying Lemma 5.1 we get

$$\begin{aligned}\hat{H}(a_1 + b_1, \ldots, a_n + b_n) &= \hat{H}((a_1 + b_1)/(A + B), \ldots, (a_n + b_n)/(A + B))\\ &\geq \min\left(\hat{H}(a_1/A, \ldots, a_n/A), \hat{H}(b_1/B, \ldots, b_n/B)\right)\\ &= \min\left(\hat{H}(a_1, \ldots, a_n), \hat{H}(b_1, \ldots, b_n)\right)\end{aligned}$$

□

Theorem 5.4. (Monotonicity of Entropy ℓ-diversity with don't-care sets) *Entropy ℓ-diversity with don't-care sets satisfies the monotone property: given a don't-care set Y, if a table $T^\star$ satisfies entropy ℓ-diversity then any generalization $T^{\star\star}$ of $T^\star$ also satisfies entropy ℓ-diversity.*

Adults

	Attribute	Domain size	Generalizations type	Ht.
1	Age	74	ranges-5,10,20	4
2	Gender	2	Suppression	1
3	Race	5	Suppression	1
4	Marital Status	7	Taxonomy tree	2
5	Education	16	Taxonomy tree	3
6	Native Country	41	Taxonomy tree	2
7	Work Class	7	Taxonomy tree	2
8	Salary class	2	*Sensitive att.*	
9	Occupation	14	*Sensitive att.*	

Lands End

	Attribute	Domain size	Generalizations type	Ht.
1	Zipcode	31953	Round each digit	5
2	Order date	320	Taxonomy tree	3
3	Gender	2	Suppression	1
4	Style	1509	Suppression	1
5	Price	346	Round each digit	4
6	Quantity	1	Suppression	1
7	Shipment	2	Suppression	1
8	Cost	147	*Sensitive att.*	

Figure 7. Description of Adults and Lands End Databases

Proof. The proof of monotonicity is an easy consequence of the following result: if q_1 and q_2 are $q^\star$-blocks, and if q_3 is the $q^\star$-block formed by merging q_1 and q_2 then the adjusted entropy of q_3 is greater than or equal to the minimum of the adjusted entropies of q_1 and q_2. Therefore, this is what we aim to prove.

Let q_1 and q_2 be $q^\star$ blocks. Let $s_1, \ldots, s_n$ be the sensitive values that appear in q_1 and q_2 and let $a_1, \ldots, a_n$ be their counts in q_1 and $b_1, \ldots, b_n$ be their counts in q_2. Let $a_i^\star$ be the values used to compute the adjusted entropy for q_1 and $b_i^\star$ be the values used to compute adjusted entropy for q_2. Note that for all i, $a_i \geq a_i^\star$ and $b_i \geq b_i^\star$. Furthermore $a_i > a_i^\star$ or $b_i > b_i^\star$ only if s_i is a don't-care value (by construction). When we merge q_1 and q_2 the new counts are $(a_i + b_i)$. By Corollary 5.1:

$$\hat{H}(a_1^\star + b_1^\star, a_2^\star + b_2^\star, \ldots, a_n^\star + b_n^\star) \geq \min\left(\hat{H}(a_1^\star, \ldots, a_n^\star), \hat{H}(b_1^\star, \ldots, b_n^\star)\right)$$

Now $a_i + b_i \geq a_i^\star + b_i^\star$ and $a_i + b_i > a_i^\star + b_i^\star$ only if s_i is a don't care value. Since the adjusted entropy is the maximum entropy we can achieve by lowering the counts associated with the don't-care values, this means that the adjusted entropy for the group with counts $a_i + b_i$ is at least $\hat{H}(a_1^\star + b_1^\star, a_2^\star + b_2^\star, \ldots, a_n^\star + b_n^\star)$. Thus the adjusted entropy of the merged group is larger than or equal to the minimum adjusted entropy of q_1 and q_2. □

Thus to create an algorithm for ℓ-diversity, we can take an algorithm for k-anonymity that performs a lattice search and we make the following change: every time a table $T^\star$ is tested for k-anonymity, we check for ℓ-diversity instead. Since ℓ-diversity is a property that is local to each $q^\star$-block and since all ℓ-diversity tests are solely based on the counts of the sensitive values, this test can be performed very efficiently.

We emphasize that this is only one way of generating ℓ-diverse tables and it is motivated by the structural similarities between k-anonymity and ℓ-diversity. Alternatively, one can post-process a k-anonymous table and suppress groups that are not ℓ-diverse or suppress tuples in groups until all groups are ℓ-diverse; one can directly modify a k-anonymity algorithm that uses suppression into an ℓ-diversity algorithm; or one can devise a completely new algorithm.

6. Experiments

In our experiments, we used an implementation of Incognito, as described in [51], for generating k-anonymous tables. We modified this implementation so that it produces ℓ-diverse tables as well. Incognito is implemented in Java and uses the database manager IBM DB2 v8.1 to store its data. All experiments were run under Linux (Fedora Core 3) on a machine with a 3 GHz Intel Pentium 4 processor and 1 GB RAM.

We ran our experiments on the Adult Database from the UCI Machine Learning Repository [61] and the Lands End Database. The Adult Database contains 45,222 tuples from US Census data and the Lands End Database contains 4,591,581 tuples of point-of-sale information. We removed tuples with missing values and adopted the same domain generalizations as [51]. Figure 7 provides a brief description of the data including the attributes we used, the number of distinct values for each attribute, the type of generalization that was used (for non-sensitive attributes), and the height of the generalization hierarchy for each attribute.

Homogeneity Attack. In Figures 8 and 9, we illustrate the *homogeneity* attacks on k-anonymized datasets using the Lands End and Adult databases. For the Lands End Database, we treated {*Zipcode, Order Date, Gender, Style, Price*} as the quasi-identifier. We partitioned the *Cost* attribute into 147 buckets by rounding to the nearest 100 and used this as the sensitive attribute. For the Adults database, we used {*Age, Gender, Race, Marital Status, Education*} as the quasi-identifier and *Salary*

Adults

k	Affected /Total tables	Avg. Gps. Affected	Avg. Tuples Affected
2	8/8	7.38	558.00
5	11/12	3.58	381.58
10	10/12	1.75	300.42
15	7/8	2.12	317.25
20	8/10	1.20	228.20
30	7/10	0.90	215.40
50	5/5	1.00	202.80

Lands End

k	Affected /Total tables	Avg. Gps. Affected	Avg. Tuples Affected
2	2/3	12.3	2537.6
5	2/3	12.3	2537.6
10	2/2	18.5	3806.5
15	2/2	18.5	3806.5
20	1/2	2.5	1750
30	1/2	2.5	1750
50	1/3	0.6	1156

Figure 8. Effect of Homogeneity Attack on the Databases

Adults

k	Affected /Total tables	Avg. Gps. Affected	Avg. Tuples Affected
2	8/8	20.50	13574.5
5	12/12	12.67	13328.3
10	12/12	7.83	10796.5
15	8/8	8.88	12009.4
20	10/10	7.10	11041.0
30	10/10	5.50	11177.0
50	5/5	5.80	8002.0

Lands End

k	Affected /Total tables	Avg. Gps. Affected	Avg. Tuples Affected
2	2/3	13.0	2825.33
5	2/3	13.0	2825.33
10	2/2	19.5	4238.00
15	2/2	19.5	4238.00
20	1/2	3.0	2119.00
30	1/2	3.0	2119.00
50	1/3	1.0	1412.66

Figure 9. Effect of 95% Homogeneity Attack on the Databases

Class as the sensitive attribute. For values of $k = 2, 5, 10, 15, 20, 30, 50$, we then generated all k-anonymous tables that were minimal with respect to the generalization lattice (i.e. no table at a lower level of generalization was k-anonymous).

Figure 8 shows an analysis of groups in k-anonymous tables that are completely homogeneous, and Figure 9 shows an analysis of groups in k-anonymous tables that are "nearly" homogeneous (i.e., the most frequent sensitive value s in a group appears in at least 95% of the tuples in the group). Both cases should be avoided since an adversary would believe, with near certainty, that an individual in a homogeneous or nearly homogeneous group has the sensitive value s that appears most frequently. Note that the minority (i.e., $\leq 5\%$) of the individuals in nearly homogeneous groups whose sensitive values are not s are also affected even though the best inference about them (that they have s) is wrong. As a concrete example, consider the case when $s = AIDS$. An individual that values privacy would not want to be associated with s with near certainty regardless of whether the true value is s.

In the four tables shown in Figures 8 and 9, the first column indicates the value of k. The second column shows the number of minimal k-anonymous tables that have groups that are completely homogeneous (Figure 8) or 95% homogenous (Figure 9). The third column shows the average number of such groups per minimal k-anonymous table. The fourth column shows the average number of tuples per minimal k-anonymous table that were affected by the two homogeneity attacks. As we can see from Figures 8 and 9, the homogeneity attack is a real concern, affecting a very large fraction of both datasets. Even for relatively large values of k (such as 30 and 50), many tables still had nearly homogeneous groups.

Note that the average number of affected groups, average number of affected tuples, etc., are not strictly decreasing functions of k. In particular, tables with small values of affected tuples are sometimes close to each other in the lattice of k-anonymous tables and may be generalized to the same table when k increases (thus reducing the total number of "safe" tables).

Performance. In our next set of experiments, we compare the running times of entropy ℓ-diversity and k-anonymity. The results are shown in Figures 10 and 11. For the Adult Database, we used *Occupation* as the sensitive attribute, and for Lands End we used *Cost*. We varied the quasi-identifier size from 3 attributes up to 8 attributes; a quasi-identifier of size j consisted of the first j attributes of its dataset as listed in Figure 7. We measured the time taken to return all 6-anonymous tables and compared it to the time taken to return all 6-diverse tables. In both datasets, the running times for k-anonymity and ℓ-diversity were similar. Sometimes the running time for ℓ-diversity was faster, which happened when the algorithm pruned parts of the generalization lattice earlier than it did for k-anonymity.

Utility. The next set of experiments compare the utility of anonymized tables which are k-anonymous, entropy ℓ-diverse, or recursive $(3, \ell)$-diverse. We use the Adults Database in all the experiments with sensitive attribute $Occupation$. For the purposes of comparison, we set $k = \ell$ and experimented with the following values of ℓ (and hence k): 2, 4, 6, 8,

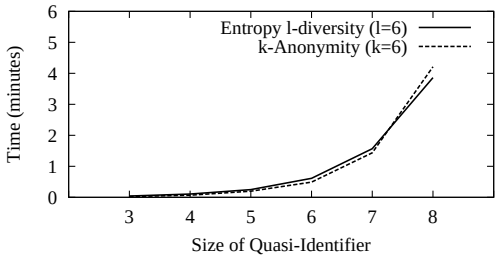

Figure 10. Adults Database

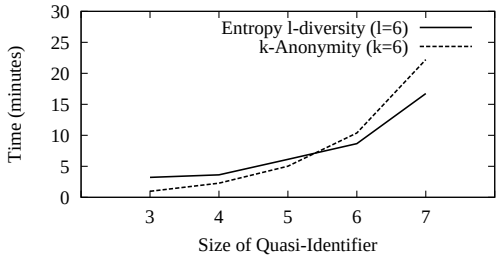

Figure 11. Lands End Database

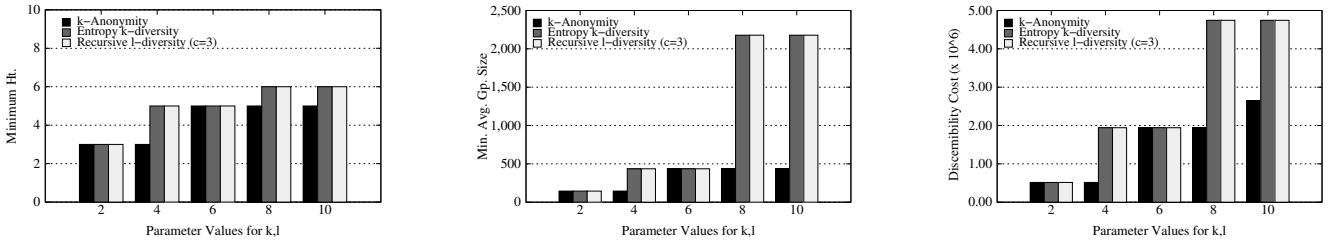

Figure 12. Adults Database. Q = {age, gender, race}

10. The sensitive attribute $Occupation$ takes only 14 values. Hence, there is no table which can be more than 14-diverse for any reasonable definition of diversity. Since some of the values appeared very infrequently, we found that there is no generalization of the Adults Database that is recursive $(3, \ell)$-diverse for $\ell = 12$. We also found that the marginal distribution of the sensitive attribute is entropy 10.57-diverse. This means that no generalization of the Adults Database can be more than entropy 10.57-diverse unless the entire data set is suppressed.

The utility of a dataset is difficult to quantify. As a result, we used four different metrics to gauge the utility of the generalized tables – generalization height, average group size, discernibility, and KL-divergence. The first metric, generalization height [51, 62], is the height of an anonymized table in the generalization lattice; intuitively, it is the number of generalization steps that were performed. The second metric is the average size of the q^*-blocks generated by the anonymization algorithm. The third metric is the *discernibility* metric [12]. The discernibility metric measures the number of tuples that are indistinguishable from each other. Each tuple in a q^* block B_i incurs a cost $|B_i|$ and each tuple that is completely suppressed incurs a cost $|D|$ (where D is the original dataset). Since we did not perform any tuple suppression, the discernibility metric is equivalent to the sum of the squares of the sizes of the q^*-blocks.

Neither generalization height, nor average group size, nor discernibility take the data distribution into account. For this reason we also use the KL-divergence, which is described next. In many data mining tasks, we would like to use the published table to estimate the joint distribution of the attributes. Now, given a table T with categorical attributes $A_1, \ldots, A_m$, we can view the data as an i.i.d. sample from an m-dimensional distribution F. We can estimate this F with the empirical distribution $\hat{F}$, where $\hat{F}(x_1, \ldots, x_m)$ is the fraction of tuples t in the table such that $t.A_1 = x_1, \ldots, t.A_m = x_m$. When a generalized version of the table is published, the estimate changes to $\hat{F}^\star$ by taking into account the generalizations used to construct the anonymized table $T^\star$ (and making the uniformity assumption for all generalized tuples sharing the same attribute values). If the tuple $t = (x_1, \ldots, x_m)$ is generalized to $t^\star = (x_1^\star, \ldots, x_m^\star)$, then $\hat{F}^\star(x_1, \ldots, x_m)$ is given by

$$\begin{aligned} \hat{F}^\star(x_1, \ldots, x_m) &= \frac{|\{t^\star \in T^\star\}|}{|T^\star| \times area(t^\star)} \\ \text{where, } area(x_1^\star, \ldots, x_m^\star) &= \prod_{i=1}^{m} |\{x_i \in A_i \mid x_i \text{ is generalized to } x_i^\star\}| \end{aligned}$$

To quantify the difference between the two distributions $\hat{F}$ and $\hat{F}^*$, we use Kullback-Leibler divergence (KL-divergence)

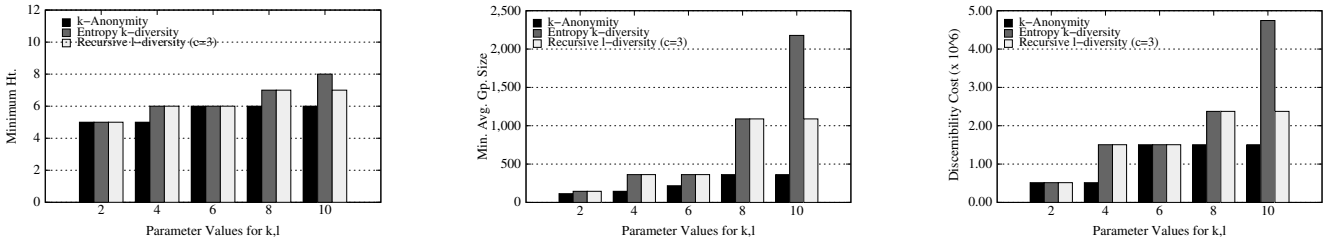

Figure 13. Adults Database. Q = {age, gender, race, marital_status}

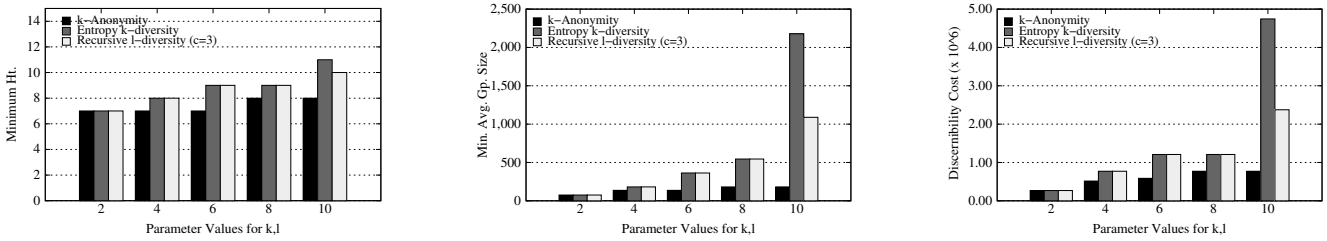

Figure 14. Adults Database. Q = {age, gender, race, marital_status, education}

which is defined as

$$\sum_{\mathbf{x} \in A_1 \times \ldots \times A_m} \hat{F}(\mathbf{x}) \log \frac{\hat{F}(\mathbf{x})}{\hat{F}^{\star}(\mathbf{x})}$$

where $0 \log 0$ is defined to be 0. The KL-divergence is non-negative and is 0 only when the two estimates are identical.

In Figures 12, 13, 14, and 15, we show the minimum generalization height, average group size, and discernibility of k-anonymous, entropy ℓ-diverse, and recursive $(3, \ell)$-diverse tables for $\ell = k = 2, 4, 6, 8, 10$, while Figures 16 and 17 show our results for KL-divergence. For each graphs in Figures 12, 13, 14, 15, and 16, we performed the anonymizations on a 5% subsample of the original data, while Figure 17 shows results for anonymization of the entire data set.

Before explaining why it was necessary to subsample the data, we should first note that in general, the graphs show that ensuring diversity in the sensitive attribute does not require many more generalization steps than for k-anonymity (note that an ℓ-diverse table is automatically ℓ-anonymous); the minimum generalization heights for identical values of k and ℓ were usually identical. Nevertheless, we found that generalization height was not an ideal utility metric because tables with small generalization heights can still have very large group sizes. For example, using full-domain generalization on the Adult Database with the quasi-identifier {*Age, Gender, Race, Marital Status, Education*}, we found minimal (with respect to the generalization lattice) 4-anonymous tables that had average group sizes larger than 1,000 tuples. The large groups were caused by data skew. For example, there were only 114 tuples with age between 81 and 90, while there were 12,291 tuples with age between 31 and 40. So if age groups of length 5 (i.e. [1-5], [6-10], [11-15], etc) were generalized to age groups of length 10 (i.e. [1-10], [11-20], etc), we would end up with very large q^*-blocks.[4]

Thus, to better understand the loss of utility due to domain generalization, we chose to study a subsample of the Adults Database with a lesser data skew in the Age attribute. It turned out that a 5% Bernoulli subsample of the Adult Database suited our requirements – most of the Age values appeared in around 20 tuples each, while only a few values appeared in less than 10 tuples each. The second and third graphs in each of Figures 12, 13, 14, and 15 show the minimum average group size and the discernibility metric cost, respectively, of k-anonymous and ℓ-diverse tables for $k, \ell = 2, 4, 6, 8, 10$. Smaller values for utility metrics represent higher utility. We found that the best t-anonymous and t-diverse tables often (but not always) had comparable utility. It is interesting to note that recursive $(3, \ell)$-diversity permits tables which have better utility than entropy ℓ-diversity. Recursive (c, ℓ)-diversity is generally less restrictive than entropy ℓ-diversity, because the extra parameter, c,

[4]Generalization hierarchies that are aware of data skew may yield higher quality anonymizations. This is a promising avenue for future work because some recent algorithms [12] can handle certain dynamic generalization hierarchies.

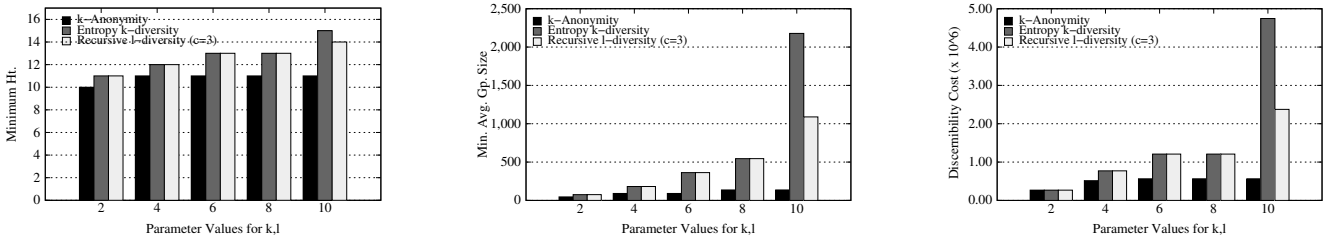

Figure 15. Adults Database. Q = {age, gender, race, marital_status, education, work_class, native_country}

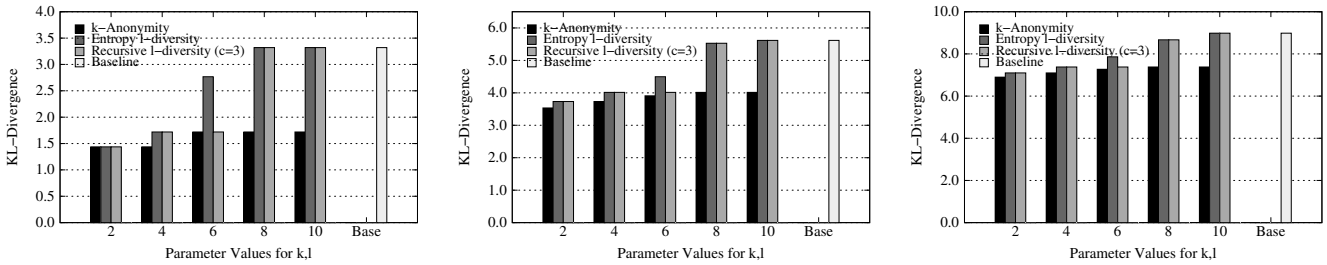

Figure 16. Comparing KL-Divergence to k-Anonymous and ℓ-Diverse versions of a sample of the Adults Database. From left to right, $Q = \{$*Age, Gender, Race*$\}$, $\{$*Age, Gender, Marital Status, Race*$\}$ and $\{$*Age, Education, Gender, Marital Status, Race*$\}$ respectively.

allows us to control how much skew is acceptable in a q^*-block. Since there is still some residual skew even in our 5% subsample, the entropy definition performs worse than the recursive definition.

In Figures 16 and 17 we compare k-anonymous and ℓ-diverse tables using the KL-divergence utility metric. Figure 16 shows our results for a 5% subsample of the table and Figure 17 shows our results on the whole Adults Database. In each of the graphs, we wish to publish a table from which the joint distribution $Q \times S$ can be estimated. In all the cases $S = Occupation$. Q is the multi-dimensional attribute $\{Age, Gender, Race\}$, $\{Age, Gender, Marital\_Status, Race\}$ and $\{Age, Education, Gender, Marital\_Status, Race\}$, respectively.

Each of the graphs shows a base-line (the bar named "Base") that corresponds to the KL-divergence for the table where all the attributes in Q were completely suppressed (thus the resulting table had only one attribute – the sensitive attribute). This table represents the least useful anonymized table that can be published. The rest of the bars correspond to the KL-divergence to the best k-anonymous, entropy ℓ-diverse, and recursive $(3, \ell)$-diverse tables, respectively for $k = \ell = 2, 4, 6, 8, 10$.

In the experiments run on the full Adults Dataset, we see that the KL-divergence to the best ℓ-diverse table (entropy or recursive) is very close to the KL-divergence to the best k-anonymous table, for $k = \ell = 2, 4, 6$. As expected, for larger values of ℓ, the utility of ℓ-diverse tables is lower. The best tables for the entropy and recursive variants of the definition often have similar utility. When a sample of Adults Database table was used, some of the sensitive values with small counts were eliminated. Hence, for $\ell = 8, 10$, the best tables were very close to the baseline. For $\ell = 6$, the recursive definition performs better than the entropy definition since recursive $(3, \ell)$-diversity allows for more skew in the sensitive attribute.

7. Related Work

There has a been a lot of research on individual data privacy in both the computer science and the statistics literature. While a comprehensive treatment is outside the scope of this paper, we provide an overview of the area by discussing representative work. Most of the work can be broadly classified depending on whether or not the data collector is trusted. We first discuss the trusted data collector scenario, of which our work is an example, in Section 7.1. We then discuss the untrusted data

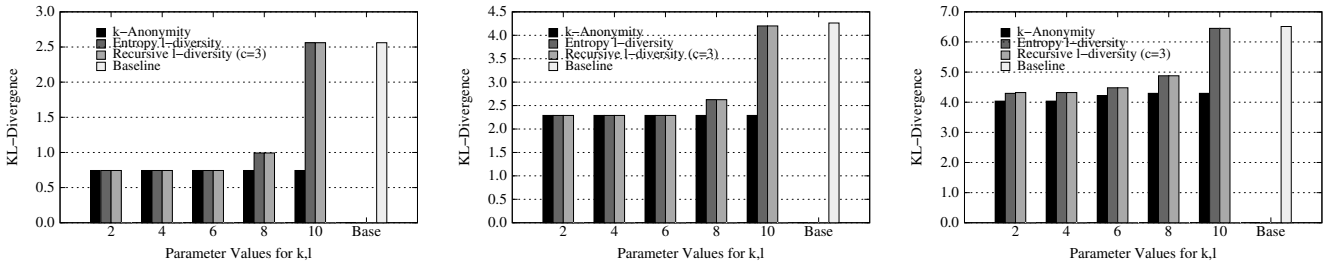

Figure 17. Comparing KL-Divergence to k-Anonymous and ℓ-Diverse versions of the Adults Database. From left to right, $Q = \{$*Age, Gender, Race*$\}$, $\{$*Age, Gender, Marital Status, Race*$\}$ and $\{$*Age, Education, Gender, Marital Status, Race*$\}$ respectively.

collector scenario in Section 7.2.

7.1. Trusted Data Collector

In many scenarios, the individuals providing the data trust the data collector not to breach their privacy. Examples of such data collectors are the Census Bureau, hospitals, health insurance providers, etc. However, these data collectors want to share data with third parties for enhancing research. It is required that such sharing does not breach the privacy of the individuals. Methods used by the data collectors can be broadly classified into four classes (each of which is discussed below):

- Publish public-use microdata (e.g., the approach taken in this paper).
- Allow third parties to query the data, and only allow queries which do not lead to disclosures (like in statistical databases).
- Share data only with authorized third parties.
- Do not share data but provide support for collaborative computations which disclose no information beyond the final answer.

7.1.1 Publishing Public-use Microdata

This paper proposes new privacy definitions for the model of *publishing public-use microdata*. The Census Bureau provides data as public-use microdata (PUMS). They use a variety of sanitization techniques to ensure privacy and utility in the dataset. Hence, there is a huge amount of research on data sanitization in the statistics community. Here again, there are many techniques which provide some utility guarantees but do not give theoretical guarantees for privacy.

Census data literature focuses on identifying and protecting the privacy of sensitive entries in contingency tables – tables of counts which represent the complete cross-classification of the data ([43, 25, 26, 27, 37, 36, 65]). A nonzero table entry is considered sensitive if it is smaller than a fixed threshold which is usually chosen in an ad-hoc manner. Two main approaches have been proposed for protecting the privacy of sensitive cells: *data swapping* and *data suppression*. The data swapping approach involves moving data entries from one cell in the contingency table to another so that the table remains consistent with a set of published marginals [29, 32, 40]. In the data suppression approach [25, 24], cells with low counts are simply deleted. Due to data dependencies caused by marginal totals that may have been previously published, additional related cell counts may also need to be suppressed. An alternate approach is to determine a *safety range* or *protection interval* for each cell [35], and to publish only those marginals which ensure that the feasibility intervals (i.e. upper and lower bounds on the values a cell may take) contain the protection intervals for all cell entries.

Computer science research has also tried to solve the privacy preserving data publishing problem. Sweeney [68] showed that publishing data sets for which the identifying attributes (keys) have been removed is not safe and may result in privacy breaches. In fact, the paper shows a real life privacy breach using health insurance records and voter registration data. To better protect the data, [68] advocates the use of a technique called k-anonymity [63] which ensures that every individual is

hidden in a group of size at least k with respect to the non-sensitive attributes. The problem of k-anonymization is NP-hard [57]; approximation algorithms for producing k-anonymous tables have been proposed [3].

Prior to this, there had been a lot of study in creating efficient algorithms for k-anonymity by using generalization and tuple suppression techniques. Samarati et al. [63] proposed a technique, using binary search, for ensuring k-anonymity through full-domain generalization techniques. Bayardo et al. [12] modeled k-anonymization as an optimization problem between privacy and utility, and proposed an algorithm similar to a frequent itemset mining algorithm. Lefevre et al. [51] extended the approach of full-domain generalization and proposed an algorithm for returning all valid k-anonymous tables. It also used techniques very similar to frequent itemset mining. Zhong et al. [74] showed how to compute a k-anonymous table without the requirement of a trusted data collector. Ohrn et al. [60] used boolean reasoning to study the effect of locally suppressing attributes on a per-tuple basis. They introduced a notion called *relative anonymization* to counter the effects of homogeneity in the sensitive attribute. One of the instantiations of relative anonymization corresponds to the definition which we named entropy ℓ-diversity. In a preliminary version of this paper, Machanavajjhala et al. [54] first introduced ℓ-diversity which, unlike k-anonymity, was aware of the distribution of values of the sensitive attributes and of the effects of background knowledge.

Condensation based approach to ensure k-anonymity [2] treat the data as points in a high-dimensional space and the technique tries to condense k nearby points into a single point.

Chawla et al. [20] proposes a formal definition of privacy for published data based on the notion of *blending in a crowd*. Here privacy of an individual is said to be protected if an adversary cannot isolate a record having attributes similar (according to a suitably chosen distance metric) to those of a given individual without being sufficiently close (according to the distance metric) to several other individuals; these other individuals are the crowd. The authors propose several perturbation and histogram-based techniques for data sanitization prior to publication. The formalization of the notion of privacy presents a theoretical framework for studying the privacy-utility trade-offs of the proposed data sanitization techniques. However, due to the heavy reliance on an inter-tuple distance measure of privacy, the proposed definition of privacy fails to capture scenarios where identification of even a single sensitive attribute may constitute a privacy breach. Also note that this privacy definition does not guarantee diversity of the sensitive attributes.

Miklau et al. [59] characterize the set of views that can be published while keeping some query answer secret. Privacy here is defined in the information-theoretic sense of perfect privacy. They show that to ensure perfect privacy, the views that are published should not be related to the data used to compute the secret query. This shows that perfect privacy is too strict as most useful views, like those involving aggregation, are disallowed.

Finally there has been some work on publishing XML documents and ensuring access control on these documents [58, 73]. Miklau et al. [58] use cryptographic techniques to ensure that only authorized users can access the published document. Yang et al. [73] propose publishing partial documents which hide sensitive data. The challenge here is that the adversary might have background knowledge which induces dependencies between branches, and this needs to be taken into account while deciding which partial document to publish.

7.1.2 Statistical Databases

The third scenario in the trusted data collector model is hosting a *query answering service*. This is addressed by the statistical database literature. In this model, the database answers only aggregate queries (COUNT, SUM, AVG, MIN, MAX) over a specified subset of the tuples in the database. The goal of a statistical database is to answer the queries in such a way that there are no positive or negative disclosures. Techniques for statistical database query answering can be broadly classified into three categories – query restriction, query auditing, data and output perturbation. Though the literature proposes a large number of techniques for ensuring privacy, only a few of the techniques are provably private against attacks except in restricted cases. Adam et al. [1] provide a very good literature survey.

The techniques in the *query restriction* category specify the set of queries that should not be answered to ensure that privacy is not breached. None of the answers to legal queries are perturbed. All of these techniques focus on the case where a query specifies an aggregate function and a set of tuples C over which the aggregation is done. The *query set size control* technique [43, 64] specifies that only those queries which access at least $|C| \geq k$ and at most $|C| \leq L - k$ tuples should be answered. Here k is a parameter and L is the size of the database. However, it was shown that snooping tools called trackers [31] can be used to learn values of sensitive attributes. The query set overlap control technique [34] disallows queries which have a large intersection with the previous queries.

Query auditing in statistical databases has been studied in detail. The query monitoring approach [34, 21] is an online version of the problem where the $(t+1)^{th}$ query is answered or not depending on the first t queries asked. The decision is

based only on the queries and not on the answers to those queries. Pure SUM queries and pure MAX queries can be audited efficiently but the mixed SUM/MAX problem is NP-hard. In the offline auditing problem [22, 21], the queries are presented all at once and the problem is to choose the maximum number of queries that can be answered. Kleinberg et al. [49] considers auditing SUM queries over boolean attributes and shows that it is co-NP hard to decide whether a set of queries uniquely determines one of the data elements. More recently, Kenthapadi et al. [48] studied the problem of simulatable auditing. This is a variant of the query monitoring approach where the decision to disallow a query can depend on the answers to the previous queries as well. The main challenge in this model is that if a query answer is denied, information could be disclosed. Hence, the solutions proposed are such that any decision (to allow or deny a query) that is made by the database can also be simulated by the adversary.

Data perturbation techniques maintain a perturbed version of the database and answer queries on the perturbed data. However, most of these techniques suffer from the problem of bias [56]; i.e., the expected value of the query answers computed using the perturbed data is different from the actual query answers computed using the original data. Fixed data perturbation techniques [69] perturb the data by adding zero-mean random noise to every data item. Such techniques have the worst problems with bias. The randomized response scheme proposed in [71] avoids this bias problem for COUNT queries on categorical attributes. Yet another technique is to replace the data with synthetic data drawn from the same empirical distribution.

Output perturbation techniques evaluate the query on the original data but return a perturbed version of the answer. Techniques here include returning answers over a sample of the database [30], rounding the answers to a multiple of a prespecified base b [28], and adding random noise to the outputs [15]. More recently, Dinur et al. [33] proved that in order to protect against an adversary who is allowed to ask arbitrarily many queries to a database, the random noise added to the answers should be at least $\Omega(\sqrt{n})$, n being the number of tuples in the database. On the positive side, they also showed a technique that provably protects against a bounded adversary who is allowed to ask only $\mathcal{T}(n) \geq polylog(n)$ queries by using additive perturbation of the magnitude $\tilde{O}(\sqrt{\mathcal{T}(n)})$. Building on this result, Blum et al. [18] proposed a framework for practical privacy called the SuLQ framework, where the number of queries an adversary is allowed to ask is sub-linear in the number of tuples in the database.

7.1.3 Sharing with Authorized Parties

Hippocratic databases [7] are a proposed design principle for building database systems which regulate the *sharing of private data with third parties*. Such a solution requires both the individuals who provide data and the databases that collect it to specify privacy policies describing the `purposes` for which the data can be used and the `recipients` who can see parts of the data. The policies are specified using a policy specification language like APPEL [52], which satisfies the P3P standard [53]. A Hippocratic database also needs other functionality, like support for maintaining audit trails [5], query rewriting for disclosure limitation [50], and support for data retention.

Snodgrass et al. [66] proposes schemes for auditing the operations of a database such that any tampering with the audit logs can be detected. Such a solution can guard against the database's manipulation of the audit logs, thus giving assurance of eventual post-breach detection.

7.1.4 Private Collaborative Computation

Private collaborative computation has been very well studied in the form of secure multiparty computation [44, 16, 19]. The problem of secure multiparty computation deals with n parties computing a common function on private inputs. Such a protocol should not disclose to the participants any information other than what is disclosed by the answer itself. Most of the early work focused on building solutions for general functions by expressing a function as a boolean circuit. However, general solutions are perceived to be communication inefficient (of the order of the square of the number of parties involved for each gate in the boolean circuit being evaluated).

Thus there has been a lot of research proposing solutions to secure multiparty computations for specific functions. Du [38] proposes various specific (secure) two-party computations problems. The commodity server model [13, 14] has been used for privately computing the scalar product of two vectors [39]. In the commodity server model, the two (or more) parties involved in the multiparty computation protocol employ the services of an untrusted third party to provide some randomness [13] or to help with some computation [39]. It is assumed that this untrusted third party does not collude with the players involved in the multiparty computation. Most of these techniques employ randomization to guarantee privacy.

Agrawal et al. [6] employ commutative encryption techniques for information sharing across private database. Their techniques can be used to calculate the intersection and equijoin of two databases while disclosing only the sizes of each

database. Clifton et al. [23] describes methods to implement basic operations like secure sum, secure set union, secure set intersection, and secure scalar product using both encryption and additive randomization in the secure multiparty computation setting. These primitives are used in various application scenarios to build multiparty protocols for private association rule mining in horizontally partitioned data [46], private association rule mining in vertically partitioned data [70], and private EM clustering.

One drawback which permeates the above literature is that there is no clear characterization of how much information is disclosed by the output of the protocol about the sensitive inputs.

7.2. Untrusted Data Collector

In the case where the data collector is not trusted, and the private information of the individuals should be kept secret from the data collector. Though this is not the model dealt with in this paper, definitions of privacy can be common across the trusted and the untrusted data collector model. The individuals provide randomized versions of their data to the data collector who then uses it for data mining. Warner [72] proposed one of the first techniques for randomizing categorical answers to survey questionnaires. Recent work in the privacy preserving data mining literature also fits this model. Agrawal et al. [9] propose randomization techniques that can be employed by individuals to mask their sensitive information while allowing the data collector to build good decision trees on the data. This work, however, does not give theoretical guarantees for privacy. Subsequent work propose metrics for quantifying the information lost and the privacy guaranteed by privacy-preserving data mining techniques. One privacy metric [4] is based on the conditional differential entropy between the original and perturbed data. However, this privacy metric measures average-case behavior, so that a perturbed distribution can leave a lot of uncertainty about the original values in most of the domain, leave very little uncertainty in a small part of the domain (therefore causing a privacy breach), and yet still be considered satisfactory based on its conditional differential entropy. Evfimievski et al. [41, 42] propose randomization techniques for privacy-preserving association rule mining and give theoretical guarantees for privacy. They define a privacy breach to be the event that the posterior probability (of certain properties of the data) given the randomized data is far from the prior probability. These techniques deal with categorical attributes only. Extensions to continuous data that allow the data collector to run OLAP-style queries on the data have also been proposed ([10]).

On the negative side, [47] shows that randomizing the data, especially by adding zero mean random variables, does not necessarily preserve privacy. The techniques provided in the paper exploit spectral properties of random matrices to remove the noise and recover the original data. Thus the data collector could breach privacy. [45] show that the correlation between attributes is the key factor behind the attacks proposed in [47]. The paper goes on to propose two techniques based on Principle Component Analysis (PCA) and the Bayes Estimate (BE) to reconstruct the original data from the randomized data. On a positive note, the paper shows that randomization schemes where the correlations in the noise are "similar" to the correlations in the data can protect against these attacks.

8. Conclusions and Future Work

In this paper we have shown theoretically and experimentally that a k-anonymized dataset permits strong attacks due to lack of diversity in the sensitive attributes. We have introduced ℓ-diversity, a framework that gives stronger privacy guarantees. We have also demonstrated that ℓ-diversity and k-anonymity have enough similarity in their structure that k-anonymity algorithms can be modified to work with ℓ-diversity.

There are several avenues for future work. First, we want to extend our initial ideas for handling multiple sensitive attributes, and we want to develop methods for continuous sensitive attributes. Second, although privacy and utility are duals of each other, privacy has received much more attention than the utility of a published table. As a result, the concept of utility is not well-understood.

Acknowledgments. We thank Joe Halpern for an insightful discussion on the proposed privacy model, we thank Kristen LeFevre for the Incognito source code, we thank Chris Clifton for first bringing the article by Ohrn et al. [60] to our attention, we thank Richard A. Suss for the reference on entropic means [17], and we thank the anonymous reviewers for their helpful suggestions. This work was partially supported by the National Science Foundation under Grant IIS-0541507, a Sloan Foundation Fellowship, and by a gift from Microsoft. 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.

References

[1] N. R. Adam and J. C. Wortmann. Security-control methods for statistical databases: A comparative study. *ACM Comput. Surv.*, 21(4):515–556, 1989.

[2] C. C. Aggarwal and P. S. Yu. A condensation approach to privacy preserving data mining. In *EDBT*, pages 183–199, 2004.

[3] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas, and A. Zhu. k-anonymity: Algorithms and hardness. Technical report, Stanford University, 2004.

[4] D. Agrawal and C. C. Aggarwal. On the design and quantifiaction of privacy preserving data mining algorithms. In *PODS*, May 2001.

[5] R. Agrawal, R. J. Bayardo, C. Faloutsos, J. Kiernan, R. Rantzau, and R. Srikant. Auditing compliance with a hippocratic database. In *VLDB*, pages 516–527, 2004.

[6] R. Agrawal, A. V. Evfimievski, and R. Srikant. Information sharing across private databases. In *SIGMOD Conference*, pages 86–97, 2003.

[7] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu. Hippocratic databases. In *VLDB*, pages 143–154, 2002.

[8] R. Agrawal and R. Srikant. Fast Algorithms for Mining Association Rules in Large Databases. In *VLDB*, 1994.

[9] R. Agrawal and R. Srikant. Privacy preserving data mining. In *Proceedings of the 19th ACM SIGMOD Conference on Management of Data*, May 2000.

[10] R. Agrawal, R. Srikant, and D. Thomas. Privacy preserving OLAP. In *Proceedings of the 23th ACM SIGMOD Conference on Management of Data*, June 2004.

[11] F. Bacchus, A. J. Grove, J. Y. Halpern, and D. Koller. From statistical knowledge bases to degrees of belief. *A.I.*, 87(1-2), 1996.

[12] R. J. Bayardo and R. Agrawal. Data privacy through optimal k-anonymization. In *ICDE-2005*, 2005.

[13] D. Beaver. Commodity-based cryptography (extended abstract). In *STOC '97: Proceedings of the 29th ACM Symposium on Theory of Computing*, pages 446–455, 1997.

[14] D. Beaver. Server-assisted cryptography. In *NSPW '98: Proceedings of the 1998 Workshop on New security paradigms*, pages 92–106, 1998.

[15] L. Beck. A security mechanism for statistical database. *ACM Transactions on Database Systems*, 5(3):316–338, 1980.

[16] M. Ben-Or, S. Goldwasser, and A. Wigderson. Completeness theorems for non-cryptographic fault-tolerant distributed computation. In *STOC '88: Proceedings of the 20th ACM Symposium on Theory of Computing*, pages 1–10, 1988.

[17] A. Ben-Tal, A. Charnes, and M. Teboulle. Entropic means. *Journal of Mathematical Analysis and Applications*, 139(2):537–551, 1989.

[18] A. Blum, C. Dwork, F. McSherry, and K. Nissim. Practical privacy: The SuLQ framework. In *PODS*, 2005.

[19] D. Chaum, C. Crepeau, and I. Damgard. Multiparty unconditionally secure protocols. In *STOC '88: Proceedings of the 20th ACM Symposium on Theory of Computing*, pages 11–19, 1988.

[20] S. Chawla, C. Dwork, F. McSherry, A. Smith, and H. Wee. Toward privacy in public databases. In *TCC*, 2005.

[21] F. Chin. Security problems on inference control for sum, max, and min queries. *J. ACM*, 33(3):451–464, 1986.

[22] F. Chin and G. Ozsoyoglu. Auditing for secure statistical databases. In *ACM 81: Proceedings of the ACM '81 conference*, pages 53–59, 1981.

[23] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, and M. Y. Zhu. Tools for privacy preserving data mining. *SIGKDD Explorations*, 4(2):28–34, 2002.

[24] L. Cox. Network models for complementary cell suppression. *Journal of the American Statistical Association*, 90:1453–1462, 1995.

[25] L. H. Cox. Suppression, methodology and statistical disclosure control. *Journal of the American Statistical Association*, 75, 1980.

[26] L. H. Cox. Solving confidentiality protection problems in tabulations using network optimization: A network model for cell suppression in the u.s. economic censuses. In *Proceedings of the International Seminar on Statistical Confidentiality*, pages 229–245, International Statistical Institute, Dublin, 1982.

[27] L. H. Cox. New results in dislosure avoidance for tabulations. In *International Statistical Institute Proceedings of the 46th Session*, pages 83–84, Tokyo, 1987.

[28] T. Dalenius. A simple procedure for controlled rounding. *Statistik Tidskrift*, 1981.

[29] T. Dalenius and S. Reiss. Data swapping: A technique for disclosure control. *Journal of Statistical Planning and Inference*, 6, 1982.

[30] D. Denning. Secure statistical databases with random sample queries. *ACM Transactions on Database Systems*, 5(3):291–315, 1980.

[31] D. E. Denning, P. J. Denning, and M. D. Schwartz. The tracker: A threat to statistical database security. *ACM Transactions on Database Systems (TODS)*, 4(1):76–96, 1979.

[32] P. Diaconis and B. Sturmfels. Algebraic algorithms for sampling from conditional distributions. *Annals of Statistics*, 1:363–397, 1998.

[33] I. Dinur and K. Nissim. Revealing information while preserving privacy. In *PODS*, pages 202–210, 2003.

[34] D. P. Dobkin, A. K. Jones, and R. J. Lipton. Secure databases: Protection against user influence. *ACM: Transactions on Database Systems (TODS)*, 4(1):76–96, March 1979.

[35] A. Dobra. *Statistical Tools for Disclosure Limitation in Multiway Contingency Tables*. PhD thesis, Carnegie Mellon University, 2002.

[36] A. Dobra and S. E. Feinberg. *Assessing the risk of disclosure of confidential categorical data*. Bayesian Statistics 7, Oxford University Press, 2000.

[37] A. Dobra and S. E. Feinberg. Bounding entries in multi-way contingency tables given a set of marginal totals. In *Foundations of Statistical Inference: Proceedings of the Shoresh Conference 2000*. Springer Verlag, 2003.

[38] W. Du. *A Study of Several Specific Secure Two-party Computation Problems*. PhD thesis, Purdue University, 2001.

[39] W. Du and Z. Zhan. A practical approach to solve secure multi-party computation problems. In *New Security Paradigms Workshop 2002*, 2002.

[40] G. T. Duncan and S. E. Feinberg. Obtaining information while preserving privacy: A markov perturbation method for tabular data. In *Joint Statistical Meetings*, Anaheim, CA, 1997.

[41] A. Evfimievski, J. Gehrke, and R. Srikant. Limiting privacy breaches in privacy preserving data mining. In *PODS*, 2003.

[42] A. Evfimievsky, R. Srikant, J. Gehrke, and R. Agrawal. Privacy preserving data mining of association rules. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery in Databases and Data Mining*, pages 217–228, July 2002.

[43] I. P. Fellegi. On the question of statistical confidentiality. *Journal of the American Statistical Association*, 67:337:7–18, 1972.

[44] O. Goldreich, S. Micali, and A. Wigderson. How to play any mental game. In *STOC '87: Proceedings of the 19th ACM Conference on Theory of Computing*, pages 218–229, 1987.

[45] Z. Huang, W. Du, and B. Chen. Deriving private information from randomized data. In *Proceedings of the 23th ACM SIGMOD Conference on Management of Data*, June 2004.

[46] M. Kantarcioglu and C. Clifton. Privacy-preserving distributed mining of association rules on horizontally partitioned data. In *DMKD*, 2002.

[47] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar. On the privacy preserving properties of random data perturbation techniques. In *ICDM*, pages 99–106, 2003.

[48] K. Kenthapadi, N. Mishra, and K. Nissim. Simulatable auditing. In *PODS*, 2005.

[49] J. Kleinberg, C. Papadimitriou, and P. Raghavan. Auditing boolean attributes. In *PODS*, 2000.

[50] K. LeFevre, R. Agrawal, V. Ercegovac, R. Ramakrishnan, Y. Xu, and D. J. DeWitt. Limiting disclosure in hippocratic databases. In *VLDB*, pages 108–119, 2004.

[51] K. LeFevre, D. DeWitt, and R. Ramakrishnan. Incognito: Efficient fulldomain k-anonymity. In *SIGMOD*, 2005.

[52] editor M. Langheinrich. A p3p preference exchange language 1.0 (appel1.0). W3C Working Draft, February 2001.

[53] editor M. Marchiori. The platform for privacy preferences 1.0 (p3p1.0) specification. W3C Proposed Recommendation, January 2002.

[54] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. ℓ-diversity: Privacy beyond k-anonymity. In *ICDE*, 2006.

[55] David Martin, Daniel Kifer, Ashwin Machanavajjhala, Johannes Gehrke, and Joseph Halpern. Worst-case background knowledge in privacy. Technical report, Cornell University, 2006.

[56] N. S. Matloff. Another look at the use of noise addition for database security. In *Proceedings of IEEE Symposium on Security and Privacy*, pages 173–180, 1986.

[57] A. Meyerson and R. Williams. On the complexity of optimal k-anonymity. In *PODS*, 2004.

[58] G. Miklau and D. Suciu. Controlling access to published data using cryptography. In *VLDB*, pages 898–909, 2003.

[59] G. Miklau and D. Suciu. A formal analysis of information disclosure in data exchange. In *SIGMOD*, 2004.

[60] Aleksander Ohrn and Lucila Ohno-Machado. Using boolean reasoning to anonymize databases. *Artificial Intelligence in Medicine*, 15(3):235–254, 1999.

[61] U.C.Irvine Machine Learning Repository. http://www.ics.uci.edu/ mlearn/mlrepository.html.

[62] P. Samarati. Protecting respondents' identities in microdata release. In *IEEE Transactions on Knowledge and Data Engineering*, 2001.

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

[64] J. Schlorer. Identification and retrieval of personal records from a statistical bank. In *Methods Info. Med.*, 1975.

[65] A. Slavkovic and S. E. Feinberg. Bounds for cell entries in two-way tables given conditional relative frequencies. In *Privacy in Statistical Databases*, 2004.

[66] R. T. Snodgrass, S. Yao, and C. S. Collberg. Tamper detection in audit logs. In *VLDB*, pages 504–515, 2004.

[67] L. Sweeney. Uniqueness of simple demographics in the u.s. population. Technical report, Carnegie Mellon University, 2000.

[68] L. Sweeney. k-anonymity: a model for protecting privacy. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10(5):557–570, 2002.

[69] J. F. Traub, Y. Yemini, and H. Wozniakowski. The statistical security of a statistical database. *ACM Transactions on Database Systems (TODS)*, 9(4):672–679, 1984.

[70] J. Vaidya and C. Clifton. Privacy preserving association rule mining in vertically partitioned data. In *KDD*, pages 639–644, 2002.

[71] S. L. Warner. Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 1965.

[72] S. L. Warner. The linear randomized response model. *Journal of American Statistical Association*, pages 884–888, 1971.

[73] X. Yang and C. Li. Secure XML publishing without information leakage in the presence of data inference. In *VLDB*, pages 96–107, 2004.

[74] S. Zhong, Z. Yang, and R. N. Wright. Privacy-enhancing k-anonymization of customer data. In *PODS*, 2005.

A. Correctness of Entropy ℓ-diversity with Don't-care Sets

In this section we will prove Theorem 4.1. Recall that we defined normalized entropy as:

$$\hat{H}(x_1, \ldots, x_n) = -\sum_{i=1}^{n} \frac{x_i}{\sum_{j=1}^{n} x_j} \log \left(\frac{x_i}{\sum_{j=1}^{n} x_j} \right) \tag{11}$$

First, we note that as a function of $x_1, \ldots, x_n$, the normalized entropy $\hat{H}(x_1, \ldots, x_n)$ is concave. However, if we fix some of the variables, then $\hat{H}$ is neither concave nor convex in the other variables. As an example, consider $f(x) = \hat{H}(x, 100)$. We see that $f(400) = .5004$, $f(800) = .3488$, and $f(600) = .4101$. Thus $f(600) = f(\frac{1}{2} \cdot 400 + \frac{1}{2} \cdot 800) \leq \frac{1}{2} f(400) + \frac{1}{2} f(800)$ showing that the normalized entropy is not concave. However, $f(75) = .6829$, $f(125) = .6870$, and $f(100) = .6931$ Thus $f(100) = f(\frac{1}{2} \cdot 75 + \frac{1}{2} \cdot 125) \geq \frac{1}{2} f(75) + \frac{1}{2} f(125)$ and so it is not convex either. Therefore we cannot use convexity arguments to prove uniqueness in Theorem 4.1.

We begin by looking at the first-order partial derivatives of $\hat{H}$ and finding the general unconstrained maximum of $\hat{H}(x_1, \ldots, x_r, p_1, \ldots, p_m)$ where the p_i are constants. Define $f(x_1, \ldots, x_r) \equiv \hat{H}(x_1, \ldots, x_r, p_1, \ldots, p_m)$. Then $f(x_1, \ldots, x_r)$ equals:

$$-\sum_{i=1}^{r} \frac{x_i}{\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j} \log \left(\frac{x_i}{\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j} \right) - \sum_{i=1}^{m} \frac{p_i}{\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j} \log \left(\frac{p_i}{\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j} \right)$$

Simple manipulation shows that:

$$f(x_1, \ldots, x_r) = -\sum_{i=1}^{r} \frac{x_i}{\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j} \log x_i - \sum_{i=1}^{m} \frac{p_i}{\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j} \log p_i + \log \left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j \right)$$

Using the fact that the first derivative of $x \log x$ is $1 + \log x$:

$$\begin{aligned}
\frac{\partial f}{\partial x_s} &= -\frac{1 + \log x_s}{\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j} + \frac{x_s \log x_s}{\left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j \right)^2} + \sum_{i \neq s} \frac{x_i \log x_i}{\left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j \right)^2} \\
&\quad + \sum_{i=1}^{m} \frac{p_i \log p_i}{\left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j \right)^2} + \frac{1}{\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j} \\
&= -\frac{\log x_s}{\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j} + \frac{x_s \log x_s}{\left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j \right)^2} + \frac{\sum_{i \neq s} x_i \log x_i}{\left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j \right)^2} + \frac{\sum_{i=1}^{m} p_i \log p_i}{\left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j \right)^2} \\
&= -\frac{\left(\sum_{i=1}^{r} x_i + \sum_{i=1}^{m} p_i \right) \log x_s}{\left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j \right)^2} + \frac{x_s \log x_s}{\left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j \right)^2} + \frac{\sum_{i \neq s} x_i \log x_i}{\left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j \right)^2} + \frac{\sum_{i=1}^{m} p_i \log p_i}{\left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j \right)^2} \\
&= \frac{\sum_{i \neq s} (x_i \log x_i - x_i \log x_s) + \sum_{i=1}^{m} (p_i \log p_i - p_i \log x_s)}{\left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j \right)^2}
\end{aligned} \tag{12}$$

and so we see that $\partial f/\partial x_s = 0$ when

$$\log x_s = \frac{\sum\limits_{i\neq s} x_i \log x_i + \sum\limits_{i=1}^{m} p_i \log p_i}{\sum\limits_{j\neq s} x_j + \sum\limits_{j=1}^{m} p_j} \tag{13}$$

We will denote the value of the right hand side of Equation 13 by c^*. From Equation 12 it is easy to see that $\partial f/\partial x_s < 0$ when $\log(x_s) > c^*$ (when $x_s > e^{c*}$) and $\partial f/\partial x_s > 0$ when $\log(x_s) < c^*$ (when $x_s < e^{c*}$). Combining this with the fact that f is continuous at $x_s = 0$ (to rule out a maximum at $x_s = 0$), we get that given $p_1, \ldots, p_m$ and for fixed $x_1, \ldots, x_{s-1}, x_{s+1}, \ldots, x_r$, there is a unique value of x_s that maximizes $\hat{H}$. This brings us to the first theorem:

Theorem A.1. *Let $p_1, \ldots, p_m$ be constants and let $x_1, \ldots, x_{s-1}, x_{s+1}, \ldots, x_r$ be fixed. Then $\hat{H}(p_1, \ldots, p_m, x_1, \ldots, x_r)$ (when treated as a function of x_s) is maximized when*

$$\log x_s = c^* = \frac{\sum\limits_{i\neq s} x_i \log x_i + \sum\limits_{i=1}^{m} p_i \log p_i}{\sum\limits_{j\neq s} x_j + \sum\limits_{j=1}^{m} p_j}$$

Furthermore, the maximum is unique and H is decreasing for $x_s > e^{c*}$ and increasing for $x_s < e^{c*}$.

Corollary A.1. *Let $p_1, \ldots, p_m$ be constants and let $x_1, \ldots, x_{s-1}, x_{s+1}, \ldots, x_r$ be fixed. Let $\phi_s > 0$. Then $\hat{H}(p_1, \ldots, p_m, x_1, \ldots, x_r)$ (when treated as a function of x_s) is maximized subject to the constraint $x_s \leq \phi_s$ when*

$$\log x_s = \min\left(\log \phi_s, \quad \frac{\sum\limits_{i\neq s} x_i \log x_i + \sum\limits_{i=1}^{m} p_i \log p_i}{\sum\limits_{j\neq s} x_j + \sum\limits_{j=1}^{m} p_j}\right) = \min(\log \phi, M(x_1, \ldots, x_{s-1}, x_{s+1}, \ldots, x_r, p_1, \ldots, p_m))$$

Proof. If x_s cannot obtain the optimal value specified in Theorem A.1, it must be because $\phi_s < e^{c*}$. Since $\partial \hat{H}/\partial x_s > 0$ for $x_s < e^{c*}$, the maximum constrained value must occur at $x_s = \phi_s$. □

Our next step is to find the unconstrained maximum of $\hat{H}$ over $x_1, \ldots, x_r$. A necessary condition for the maximum is that all first partial derivatives are 0. From Equation 13 we have:

$$\begin{aligned}
\left(\sum_{j\neq s} x_j + \sum_{j=1}^{m} p_j\right) \log x_s &= \sum_{i\neq s} x_i \log x_i + \sum_{i=1}^{m} p_i \log p_i \\
\left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j\right) \log x_s &= \sum_{i=1}^{r} x_i \log x_i + \sum_{i=1}^{m} p_i \log p_i
\end{aligned}$$

and since the right hand side is independent of s, and since the equality is true for any s, it follows that for $s \neq t$:

$$\left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j\right) \log x_s = \left(\sum_{j=1}^{r} x_j + \sum_{j=1}^{m} p_j\right) \log x_t \tag{14}$$

$$x_s = x_t \tag{15}$$

Thus there is only one critical point and at the critical point $x_1 = x_2 = \cdots = x_r$. To find out what this value is, we go back

to Equation 13 and replace the x_i by their common value x:

$$\begin{aligned}
\log x &= \frac{(r-1)x\log x + \sum_{i=1}^{m} p_i \log p_i}{(r-1)x + \sum_{j=1}^{m} p_j} \\
(r-1)x\log x + \sum_{j=1}^{m} p_j \log x &= (r-1)x\log x + \sum_{i=1}^{m} p_i \log p_i \\
x &= \frac{\sum_{i=1}^{m} p_i \log p_i}{\sum_{j=1}^{m} p_j}
\end{aligned}$$

and we see that this is the log-entropic mean of the p_i.

Theorem A.2. *$f(x_1,\ldots,x_r) \equiv H(p_1,\ldots,p_m,x_1,\ldots,x_r)$ achieves its unique maximum when $\log x_1 = \log x_2 = \cdots = \log x_r = \frac{\sum_{i=1}^{m} p_i \log p_i}{\sum_{j=1}^{m} p_j} = c^*$.*

Proof. We have already shown that this is the unique point where all first partial derivatives are 0 at this point. We still have to show that it is that it is a global maximum. First note that a maximum cannot occur when any of the x_s are 0 (this follows directly from Theorem A.1).

Now suppose the point $(e^{c*},\ldots,e^{c*})$ is not a unique global maximum. Then there exist positive numbers $\xi_1,\xi_2,\ldots,\xi_r$ (not all equal to c^*) such that $f(\xi_1,\xi_2,\ldots,\xi_r) \geq f(e^{c*},\ldots,e^{c*})$. Let $L = \min\{p_1,\ldots,p_m,\xi_1,\ldots,\xi_r\}$ and let $U = \max\{p_1,\ldots,p_m,\xi_1,\ldots,\xi_r\}$. Consider the compact hypercube $\mathcal{C} = \{(z_1,\ldots,z_r) : \forall i \in \{1,\ldots,r\}, U \geq z_i \geq L\}$. $\mathcal{C}$ is compact, f is continuous, and f achieves its maximum on $\mathcal{C}$. Hence, there exists a point $(\theta_1,\ldots,\theta_r) \in \mathcal{C}$ such that $f(\theta_1,\ldots,\theta_r) = \sup_{z\in\mathcal{C}} f(z) \geq f(\xi_1,\ldots,\xi_r) \geq f(e^{c*},\ldots,e^{c*})$ and that not all θ_i are equal to c^*.

Now, the θ_i cannot satisfy Equation 13 (with the x_i replaced by the θ_i) for all i because otherwise we will have a second point where all the partial derivatives are 0 (a contradiction). Without loss of generality, suppose θ_1 does not satisfy Equation 13. By Theorem A.1, there exists a θ^* such that $\log\theta^*$ is a weighted average of the $\log p_i$ and $\log\theta_j$ so that $\min(p_1,\ldots,p_m,\theta_1,\ldots,\theta_r) \leq \theta^* \leq \max(p_1,\ldots,p_m,\theta_1,\ldots,\theta_r)$. This implies that $(\theta^*,\theta_2,\ldots,\theta_r) \in \mathcal{C}$. Furthermore, by Theorem A.1, $f(\theta^*,\theta_2,\ldots,\theta_r) > f(\theta_1,\ldots,\theta_r)$, which contradicts the fact that $f(\theta_1,\ldots,\theta_r)$ is maximal on $\mathcal{C}$. Therefore there do not exist any nonnegative real numbers $\xi_1,\xi_2,\ldots,\xi_r$ be nonnegative real numbers (not all equal to c^*) such that $f(\xi_1,\xi_2,\ldots,\xi_r) \geq f(e^{c*},\ldots,e^{c*})$.

□

Now that we know what the unconstrained maximum looks like, we are ready to characterize the constrained maximum. We will need the following simple results about weighted averages:

Lemma A.1. *Let $c_1,\ldots,c_n$ be nonnegative numbers and let $w_1,\ldots,w_n$ be nonnegative numbers such that $w_i c_i > 0$ for some i. Let d and v be any positive numbers.*

1. *if d equals the weighted average of the c_i (i.e., $d = (\sum_i c_i w_i)/(\sum_i w_i)$) then including d in that weighted average does not change its value (i.e., $d = (vd + \sum_i c_i w_i)/(v + \sum_i w_i) = (\sum_i c_i w_i)/(\sum_i w_i)$)*
2. *if $d > (\sum_i c_i w_i)/(\sum_i w_i)$ then $d > (vd + \sum_i c_i w_i)/(v + \sum_i w_i) > (\sum_i c_i w_i)/(\sum_i w_i)$*
3. *if $d < (\sum_i c_i w_i)/(\sum_i w_i)$ then $d < (vd + \sum_i c_i w_i)/(v + \sum_i w_i) < (\sum_i c_i w_i)/(\sum_i w_i)$*
4. *if $d > d'$ and $d > (\sum_i c_i w_i)/(\sum_i w_i)$ then $d > (vd' + \sum_i c_i w_i)/(v + \sum_i w_i)$*
5. *if $d > (vd + \sum_i c_i w_i)/(v + \sum_i w_i)$ then $d > (\sum_i c_i w_i)/(\sum_i w_i)$*

Proof. First we show (i).

$$\frac{vd + \sum_i c_i w_i}{v + \sum_i w_i} \quad = \quad \frac{vd + d\sum_i w_i}{v + \sum_i w_i} = \frac{d\,(v + \sum_i w_i)}{v + \sum_i w_i} = d = \frac{\sum_i c_i w_i}{\sum_i w_i}$$

To prove (ii), let $d^* = (\sum_i c_i w_i)/(\sum_i w_i)$ then

$$d = \frac{vd + d\sum_i w_i}{v + \sum_i w_i} > \frac{vd + \sum_i c_i w_i}{v + \sum_i w_i} > \frac{vd^* + \sum_i c_i w_i}{v + \sum_i w_i} = \frac{\sum_i c_i w_i}{\sum_i w_i}$$

and (iii) is proven the same way. (iv) is an easy consequence of (ii). To prove (v), multiply by $(v + \sum_i w_i)$ and cancel dv from both sides. □

Now we can prove the correctness of Algorithm 1 by proving Theorem 4.1, which we now restate.

Theorem A.3. *Let $p_1, \ldots, p_m, \phi_1, \ldots, \phi_r$ be positive numbers. Then the following are true:*

1. *There is a unique vector $(c_1, c_2, \ldots, c_r)$ such that the assignment $x_i = c_i$ maximizes $\hat{H}(x_1, \ldots, x_r, p_1, \ldots, p_m)$ subject to the constraints $0 \le x_i \le \phi_i$.*
2. *Let $\theta = \max(\{\phi_i \mid c_i = \phi_i\} \cup \{0\})$. If $\phi_j \le \theta$ then $c_j = \phi_j$. If $\phi_j > \theta$ then $\log c_j$ is the log-entropic mean of the set $\{p_1, \ldots, p_m\} \cup \{\phi_i \mid \phi_i = c_i\}$, and θ is the minimum value for which this condition can be satisfied.*

Proof. First we must show that a maximum exists, and this follows from the fact that $\hat{H}$ is continuous and that the set $\{(x_1, \ldots, x_r) \mid \forall i, 0 \le x_i \le \phi_i\}$ is compact. Note that uniqueness of the maximum follows from the minimality condition for θ in (ii). Therefore if we prove (ii) then (i) follows.

Let $(\xi_1, \ldots, \xi_r)$ be a point at which the maximum occurs. As a result of Corollary A.1, for $s = 1, \ldots, r$ we must have

$$\log \xi_s = \min(\log \phi, M(\xi_1, \ldots, \xi_{s-1}, \xi_{s+1}, \ldots, \xi_r, p_1, \ldots, p_m)) \tag{16}$$

Now let $W = \{i : \xi_i < \phi_i\}$ and $V = \{i : \xi_i = \phi_i\}$. We claim that:

$$\forall s \in W, \quad \log \xi_s = \frac{\sum\limits_{i \ne s} \xi_i \log \xi_i + \sum\limits_{i=1}^{m} p_i \log p_i}{\sum\limits_{j \ne s} \xi_j + \sum\limits_{j=1}^{m} p_j} = \frac{\sum\limits_{i \in V} \xi_i \log \xi_i + \sum\limits_{i=1}^{m} p_i \log p_i}{\sum\limits_{j \in V} \xi_j + \sum\limits_{j=1}^{m} p_j} \tag{17}$$

The first equality follows from Equation 16 and the second follows from Theorem A.2 for the unconstrained maximum of $\hat{H}$ as a function of x_s for $s \in W$.

Now we are ready to prove that there exists a cutoff value $\theta \in \{\phi_1, \ldots, \phi_r, 0\}$ such that $\phi_j \le \theta$ implies that $j \in V$ (i.e. $x_j = \phi_j$) and $\phi_j > \theta$ implies $j \in W$ (i.e. x_j is the log-entropic mean of the p_i and the x_s for $s \in V$). If either V or W is empty then this is trivially true. Otherwise, assume by way of contradiction that there is no cutoff so that we can find an s, t such that $\phi_s > \phi_t$ but $t \in W$ and $s \in V$. This implies that

$$\log \xi_s = \log \phi_s > \log \phi_t > \log \xi_t = M(\xi_1, \ldots, \xi_{t-1}, \xi_{t+1}, \ldots, \xi_r, p_1, \ldots, p_m))$$

and by Lemma A.1, parts (iv) and then (v), we have:

$$\log \xi_s > M(\xi_1, \ldots, \xi_r, p_1, \ldots, p_m))$$

and

$$\log \xi_s > M(\xi_1, \ldots, \xi_{s-1}, \xi_{s+1}, \ldots, \xi_r, p_1, \ldots, p_m))$$

However, this violates the condition on optimality described in Equation 16, which is a contradiction, and so there exists a cutoff θ.

All that remains to be shown is that for the optimal solution, θ is the minimum value $\in \{\phi_1, \ldots, \phi_r\}$ such that $\phi_j > \theta$ implies $j \in W$ (i.e. x_j is the log-entropic mean of the p_i and the x_s for $s \in V$). Suppose it is not minimal. Then there exists a $\theta' \in \{\phi_1, \ldots, \phi_r, 0\}$ with $\theta' < \theta$, a set $V' = \{i \mid \phi_i \le \theta'\}$ and a vector $(\omega_1, \ldots, \omega_r)$ such that when $i \in V'$ then $\omega_i = \phi_i$

and when $i \notin V'$ then ω_i is the log-entropic mean of the p_i and the ω_s for $s \in V'$. Now clearly $V' \subset V$ so whenever $\omega_i = \phi_i$ then $\xi_i = \phi_i$. However, if we fix $x_i = \phi_i$ for $i \in V'$ then the unconstrained maximum of $\hat{H}$ over the variables $\{x_i \mid i \notin V'\}$ occurs precisely when $x_i = \omega_i$, by Theorem A.2, because ω_i equals the log-entropic mean of the p_i and the ω_s for $s \in V'$. Since the variables x_s for $s \in V'$ will be fixed for any choice of cutoff θ (remember that by definition $\theta \geq \theta'$), and the unconstrained maximum over the rest of the variables is unique and achievable, the vector $(\omega_1, \ldots, \omega_r)$ that is determined by the minimal cutoff θ' is indeed the unique constrained maximum we are looking for. □