Limiting Privacy Breaches in Privacy Preserving Data Mining [*]

Alexandre Evfimievski
Cornell University
aevf@cs.cornell.edu

Johannes Gehrke
Cornell University
johannes@cs.cornell.edu

Ramakrishnan Srikant
IBM Almaden Research Center
srikant@almaden.ibm.com

ABSTRACT

There has been increasing interest in the problem of building accurate data mining models over aggregate data, while protecting privacy at the level of individual records. One approach for this problem is to randomize the values in individual records, and only disclose the randomized values. The model is then built over the randomized data, after first compensating for the randomization (at the aggregate level). This approach is potentially vulnerable to privacy breaches: based on the distribution of the data, one may be able to learn with high confidence that some of the randomized records satisfy a specified property, even though privacy is preserved on average.

In this paper, we present a new formulation of privacy breaches, together with a methodology, “amplification”, for limiting them. Unlike earlier approaches, amplification makes it is possible to guarantee limits on privacy breaches without any knowledge of the distribution of the original data. We instantiate this methodology for the problem of mining association rules, and modify the algorithm from [9] to limit privacy breaches without knowledge of the data distribution. Next, we address the problem that the amount of randomization required to avoid privacy breaches (when mining association rules) results in very long transactions. By using pseudorandom generators and carefully choosing seeds such that the desired items from the original transaction are present in the randomized transaction, we can send just the seed instead of the transaction, resulting in a dramatic drop in communication and storage cost. Finally, we define new information measures that take privacy breaches into account when quantifying the amount of privacy preserved by randomization.

1. INTRODUCTION

The explosive progress in networking, storage, and processor technologies is resulting in an unprecedented amount of digitization of information. In concert with this dramatic and escalating increase in digital data, concerns about privacy of personal information have emerged globally [6, 8, 15]. The concerns over massive collection of data are naturally extending to analytic tools applied to data. Data mining, with its promise to efficiently discover valuable, non-obvious information from large databases, is particularly vulnerable to misuse [5, 7, 15, 20].

The concept of privacy-preserving data mining has been recently been proposed in response to the above concerns [3, 13]. There have been two broad approaches. The randomization approach focuses on individual privacy, and reveals randomized information about each record in exchange for not having to reveal the original records to anyone [1, 3, 10, 18]. In the secure multi-party computation approach, the goal is to build a data mining model across multiple databases without revealing the individual records in each database to the other databases [13, 12, 21]. In this paper, we focus on privacy breaches in the context of the randomization approach. We now describe prior work in this area.

Randomization Approach The problem of building classification models over randomized data was addressed in [4, 1]. Each client has a numerical attribute x_i, e.g. age, and the server wants to learn the distribution of these attributes in order to build a classification model. The clients randomize their attributes x_i by adding random distortion values r_i drawn independently from a known distribution such as a uniform distribution over a segment or a Gaussian distribution. The server collects the values of $x_i + r_i$ and reconstructs the distribution of the x_i's using a version of the Expectation Maximization (EM) algorithm that provably [1] converges to the maximum likelihood estimate of the desired original distribution.

In [17, 9], the goal is to discover association rules over randomized data. Each client has a set of items (called a *transaction*), e.g. product preferences, and here the server wants to determine all itemsets whose support (frequency of being a subset of a transaction) is equal to or above a certain threshold. To preserve privacy, the transactions are randomized by discarding some items and inserting new items, and then are transmitted to the server. Statistical estimation of original supports and variances given randomized supports allows the server to adapt Apriori algorithm [2] to mining itemsets frequent in the non-randomized transactions by looking at only randomized ones.

Privacy It is not enough to simply concentrate on randomization and recovery of the model. We must also ensure that the randomization is sufficient for preserving *privacy*, as we randomized in the first place to achieve privacy. For example, suppose we randomize age x_i by adding a random number r_i drawn uniformly from a segment $[-50, 50]$. Assuming that the server receives age 120 from a user, privacy is somewhat compromised, as the server can conclude that the real age of the user cannot be less than 70 (otherwise $x_i + r_i < 70 + 50 = 120$). Thus the server has learned a potentially valuable piece of information about the client — information that is correct with 100% probability. Analogously, sup-

[*] This work was supported in part by NSF Grants IIS-0084762, IIS-0121175, IIS-0133481, and CCR-0205452, the Cornell Information Assurance Institute, and by gifts from Microsoft and Intel.

PODS 2003, June 9-12, 2003, San Diego, CA
Copyright 2003 ACM 1-58113-670-6/03/06 ...$5.00.

pose we randomize a small set of items (a transaction) by replacing each item by a random item with probability 80%. If the transaction contains a subset A of 3 items that has a support of 1%, it has $(0.2)^3 = 0.008 = 0.8\%$ chance to retain the same set of three items after the randomization. Thus whenever the server sees A in the randomized transaction, it learns with high probability of the presence of A in the original transaction as well. Indeed, there are $1\% \cdot 0.008 = 0.008\%$ randomized transactions that have A both before and after randomization, while the probability that A occurs in 10 randomly inserted items (out of, say, 10,000 possible items) is less than $10^{-7}\%$ [9].

We are aware of two approaches for quantifying how privacy-preserving a randomization method is. One approach relies on *information theory* [1], the other approach is based on the notion of *privacy breaches* [9]. The former approach measures the average amount of information disclosed in a randomized attribute by computing the mutual information between the original and the randomized distribution. The latter approach is a worst-case notion, it gives a criterion that should be satisfied by any privacy-preserving algorithm. Intuitively, a *privacy breach* occurs if a property of the original data record gets revealed if we see a certain value of the randomized record. In our previous example, the randomized age of 120 is an example of a privacy breach as it reveals that the actual age is at least 70. As another example, a privacy breach occurs if a subset within a randomized transaction makes it likely that some item occurs in the original transaction.

As we show in this paper, these two approaches are different: Privacy breaches can occur even though mutual information is small, and therefore propose other information-theoretical measures, called “worst-case information,” that do bound privacy breaches.

Paper Outline We introduce some basic notation in Section 2, followed by an overview of the contributions of the paper. We define privacy breaches in Section 3, and show how the amplification methodology can limit privacy breaches in Section 4. In Section 5, we use pseudorandom generators to dramatically reduce communication and storage cost of randomized transactions. We present new information measures that take privacy breaches into account in Section 6. We conclude with a summary and directions for future work in Section 7.

2. OVERVIEW

2.1 Basic Notions

The Model Suppose there are N clients $C_1, \ldots, C_N$ connected to one server; each client C_i has some private information x_i. The server needs to learn certain aggregate (statistical) properties of the clients’ data. The clients are comfortable with this, but they are reluctant to disclose their personal information x_i. To ensure privacy, each client C_i sends to the server a modified version y_i of x_i. The server collects the modified information from all clients and uses it to recover the statistical properties it needs.

Assumptions We assume that each client’s piece x_i of private information belongs to the same fixed *finite* set V_X. Furthermore, we assume that each x_i is chosen *independently at random* according to the same fixed probability distribution. This distribution, denoted p_X, is not private, the clients allow the server to learn it. The assumption of independence implies that, once p_X is known, the private information x_j of all clients C_j besides client C_i tells nothing new about C_i’s own private information x_i.

Randomization Before sending it to the server, each client C_i hides its personal data x_i by applying a *randomization operator* $R(x)$. The output of $R(x_i)$ is random, whose distribution depends on x_i and on nothing else. Only one instance y_i of $R(x_i)$ is sent to the server by client C_i. The set of all possible outputs of $R(x)$ is denoted by V_Y and is assumed to be finite. For all $x \in V_X$ and $y \in V_Y$, the probability that $R(x)$ outputs y is denoted by

$$p\,[x \to y] \;:=\; \mathbf{P}\,[R(x) = y].$$

By receiving y_i from C_i, the server learns something about x_i. Note that, by independence assumption above, all y_j for $j \neq i$ disclose nothing about x_i and can be ignored in privacy analysis; they certainly help the server to learn distribution p_X, but for privacy analysis we assume that the server knows p_X. The problem is to measure how much can be disclosed by y_i about x_i, and to find randomization operators that keep the disclosure limited.

2.2 Contributions

Refined Definition of Privacy Breaches A privacy breach is a situation when, for some client C_i, the disclosure of its randomized private information y_i to the server reveals that a certain property of C_i’s private information holds with high probability. Privacy breaches were defined in [9]; here we refine that definition by explicitly setting the limit to *prior probability* of a property. Prior probability is the likelihood of the property in the absence of any knowledge about C_i’s private information; posterior probability is the likelihood of the property given the randomized value y_i. Without a bound on prior probability, there always are properties whose posterior probability is very high even if no information is disclosed, e. g. the property $Q(x) \equiv$ “$x = x$”, In Section 3, we give the new definition (Definition 1) of privacy breaches and then further classify them into upward and downward privacy breaches. We give an example for both kinds of breaches.

Amplification Section 4.1 develops a new approach that allows to ensure limitations on privacy breaches for a randomization operator, without any knowledge about the prior distribution p_X and applicable to any property of client’s private information. Our privacy preserving restriction involves only the operator’s transition probabilities $p\,[x \to y]$:

$$\forall\, x_1, x_2 \in V_X : \;\frac{p\,[x_1 \to y]}{p\,[x_2 \to y]} \;\leqslant\; \gamma \tag{1}$$

(see Definition 3 for details). In Statement 1 we prove that if a randomization operator satisfies this condition for some randomized value y, then the disclosure of y to the server has a limited effect at breaching privacy, depending on the value of γ.

Itemset Randomization In Section 4.2 we apply amplification (Statement 1) to randomizing itemsets (in the framework of mining association rules). We give a heuristic, based on the solution of an optimization problem, that allows us to choose randomization parameters so that

- the randomization operator satisfies condition (1);
- the supports of the original itemsets can be recovered from randomized transactions.

We illustrate the practical utility of our method through some trade-off charts.

Compression of Randomized Transactions Both in the earlier approaches and in the amplification approach for itemset randomization, the randomized transactions may be very long and memory-consuming. Each randomized transaction often contains many thousands of items (order of magnitude more than original transactions); this is needed in order to hide the true items, for preserving privacy. Fortunately, there is a way to “compress” randomized transactions without compromising privacy or support recovery. The idea is to use a *pseudorandom generator* for computing which items belong to each randomized transaction. The seed for the pseudorandom generator, one seed per transaction, is chosen so that the randomized transaction contains or does not contain certain pre-selected items from the original transaction. This seed is sufficient to compute the whole randomized transaction or any portion of it, so it serves as a "compressed" randomized transaction.

Section 5 explains how one can construct a suitable pseudorandom generator using error-correcting codes. The method can reduce the size of randomized transactions by several orders of magnitude, without any effect on either privacy or support recovery. The use of the pseudorandom generator results in dropping the full probabilistic independence of "false" items inserted into the randomized transaction, but instead having only q-wise independence for a sufficiently large integer q. Privacy preserving capability of the new randomization operator can be evaluated using amplification.

Worst-Case Information In Section 6, we elaborate upon the work in [1] on measures of privacy. We show that the value of the classical mutual information does not ensure safety from privacy breaches, and introduce new information-theoretic privacy measures whose values provably bound privacy breaches. It turns out that two different subclasses of privacy breaches called "upward" and "downward" privacy breaches (Definition 2) are bounded by different measures, though measures are defined in a very similar way. Worst-case information is obtained from mutual information by writing it in terms of the Kullback-Leibler distance and replacing the expectation with the maximum.

3. PRIVACY BREACHES

Let C_i be any client, let x_i be its private information. For the server, prior to randomization, each possible value x of C_i's private information has probability $p_X(x)$ (see Section 2.1). Let us define a random variable X such that

$$\mathbf{P}[X = x] \;:=\; p_X(x).$$

Random variable X is the best description of the server's prior knowledge about x_i. Now, suppose that the client randomizes x_i by computing $y_i = R(x_i)$, then sends y_i to the server. From the server's point of view, the randomized value y_i is an instance of a random variable Y such that

$$\mathbf{P}[Y = y] \;:=\; \sum_{x \in V_X} \mathbf{P}[X = x] \cdot p\,[x \to y].$$

Random variables X and Y are dependent; their joint distribution is given by:

$$\mathbf{P}[X = x,\; Y = y] \;=\; p_X(x) \cdot p\,[x \to y].$$

Given y_i, the server can better evaluate the probabilities of possible values for C_i's private information. It uses Bayes formula and computes posterior probabilities:

$$\mathbf{P}[X = x \mid Y = y_i] \;:=\; \frac{\mathbf{P}[X = x] \cdot p\,[x \to y_i]}{\mathbf{P}[Y = y_i]}.$$

We can also find the posterior probability of any *property* $Q(x)$, where $Q: V_X \to \{\text{true},\ \text{false}\}$:

$$\mathbf{P}[Q(X) \mid Y = y_i] \;=\; \sum_{Q(x),\, x \in V_X} \mathbf{P}[X = x \mid Y = y_i].$$

Informally, a privacy breach is a situation when, for some property $Q(x)$, the disclosure of y_i to the server significantly increases the probability of this property. If it is important to the client that property $Q(x_i)$ of its private information is not disclosed, then a significant increase in probability may be a violation of privacy. Here is the formal definition of a privacy breach:

Given:	$X = 0$	$X \notin \{200, \ldots, 800\}$
nothing	1%	$\approx$ 40.5%
$R_1(X) = 0$	$\approx$ 71.6%	$\approx$ 83.0%
$R_2(X) = 0$	$\approx$ 4.8%	100%
$R_3(X) = 0$	$\approx$ 2.9%	$\approx$ 70.8%

Table 1: Prior and posterior (given $R(X) = 0$) probabilities for properties in Example 1

Definition 1. *We say that there is a ρ_1-to-ρ_2* privacy breach *with respect to property $Q(x)$ if for some $y \in V_Y$*

$$\mathbf{P}[Q(X)] \leqslant \rho_1 \quad \textit{and} \quad \mathbf{P}[Q(X) \mid Y = y] \geqslant \rho_2.$$

Here $0 < \rho_1 < \rho_2 < 1$ *and* $\mathbf{P}[Y = y] > 0$.

Let us consider the following example on privacy breaches.

Example 1. Suppose that private information x is a number between 0 and 1000. This number is chosen as a random variable X such that 0 is 1%-likely whereas any non-zero is only about 0.1%-likely:

$$\begin{aligned}\mathbf{P}[X = 0] &= 0.01\\ \mathbf{P}[X = k] &= 0.00099, \quad k = 1 \ldots 1000\end{aligned}$$

Suppose we want to randomize such a number by replacing it with a new random number $y = R(x)$ that retains some information about the original number x. Here are three possible ways to do it:

1. Given x, let $R_1(x)$ be x with 20% probability, and some other number (chosen uniformly at random) with 80% probability.
2. Given x, let $R_2(x)$ be $x + \xi \pmod{1001}$, where ξ is chosen uniformly at random in $\{-100, \ldots, 100\}$.
3. Given x, let $R_3(x)$ be $R_2(x)$ with 50% probability, and a uniformly random number otherwise.

In Table 1 we compute prior and posterior probabilities of two properties of X: property $Q_1(X) \equiv$ "$X = 0$" and property $Q_2(X) \equiv$ "$X \notin \{200, \ldots, 800\}$." We can see that randomization operator R_1 reveals a lot of information about X when $R_1(X)$ happens to equal zero: the server learns with high probability that X originally was zero. Without knowing that $R_1(X) = 0$, the server considers $X = 0$ to be just 1%-likely; but when $R_1(X) = 0$ is revealed, $X = 0$ becomes about 70%-likely. This does not happen when $R_2(X) = 0$ is revealed, the probability of $X = 0$ becomes only 4.8%. However, a different kind of personal information breaks through: the server knows with 100% *certainty* that X does not lie between 200 and 800. The prior probability of this property is about 40%. Only R_3 seems to be a good privacy preserving randomization.

As Example 1 shows, some randomization operators may not be safe because, if they are used, learning a randomized value sometimes significantly affects posterior probabilities for certain properties of the original private value. To fix this, we either have to make sure that all involved properties are harmless when disclosed to the server, or that no property significantly changes its posterior probability. In this paper we take the latter approach. According to Definition 1, for $R_1(x)$ we have a 1%-to-70% privacy breach with respect to property $Q_1(x)$, and for $R_2(x)$ we have a 40%-to-100% privacy breach with respect to property $Q_2(x)$.

What changes in probability should we classify as "significant"? In Example 1 there are two kinds of changes:

1. Some property $Q_1(x)$ has very low prior probability (i.e., is *unlikely*), but becomes *likely* once we learn that $R(X) = y$. In Example 1, the property $X = 0$ has a probability jump from 1% to above 70% when $R_1(X) = 0$ is revealed.
2. Some property $Q_2(x)$ has a probability far from 100% (i.e., is *uncertain*), but becomes almost 100%-probable (i.e., almost *certain*) once we learn that $R(X) = y$. Another way of looking at it is by taking a negation: property $\neg Q_2(X)$ is likely, but becomes very unlikely once $R(X) = y$ is revealed. In Example 1, the property "$200 \leqslant X \leqslant 800$" has a downward probability jump from almost 60% to 0% when $R_2(X) = 0$ is revealed, making it certain that either $X < 200$ or $X > 800$.

This observation suggests that there are two important subclasses of privacy breaches. Let us now give the formal definitions for both of these subclasses. Let ρ_1 and ρ_2 be two probabilities, such that ρ_1 corresponds to our intuitive notion of "very unlikely" (e.g., $\rho_1 = 0.01$) whereas ρ_2 corresponds to "likely" (e.g., $\rho_2 = 0.5$); let $Q_1(x)$ and $Q_2(x)$ be two properties.

Definition 2. *We say that there is a* straight *or* upward *ρ_1-to-ρ_2 privacy breach with respect to Q_1 if for some $y \in V_Y$*

$$\mathbf{P}[Q_1(X)] \leqslant \rho_1, \quad \mathbf{P}[Q_1(X) \mid R(X) = y] \geqslant \rho_2.$$

We say that there is an inverse *or* downward *ρ_2-to-ρ_1 privacy breach with respect to Q_2 if for some $y \in V_Y$*

$$\mathbf{P}[Q_2(X)] \geqslant \rho_2, \quad \mathbf{P}[Q_2(X) \mid R(X) = y] \leqslant \rho_1.$$

Using property $Q_2' = \neg Q_2$, we could write this as

$$\mathbf{P}[Q_2'(X)] \leqslant 1 - \rho_2, \quad \mathbf{P}[Q_2'(X) \mid R(X) = y] \geqslant 1 - \rho_1.$$

We also say that the breach is caused *by the value $y \in V_Y$ from the inequalities; we assume that $\mathbf{P}[R(X) = y] > 0$.*

So, the probability of $1 - \rho_2$ corresponds to the intuitive notion of "uncertain," and the probability of $1 - \rho_1$ means "almost certain." Our task in the next section is to define sufficient conditions for randomization operators that guarantee no breaches of either kind for any property (for given ρ_1 and ρ_2), regardless of the prior distribution p_X. Then we shall look at the problem of constructing the operators that satisfy these conditions and still allow aggregate data mining.

4. AMPLIFICATION

If we attempt to use Definition 1 directly to check whether a given randomization operator R causes privacy breaches, we immediately encounter two difficulties:

1. There are $2^{|V_X|}$ possible properties, far too many to check them all;
2. We cannot use Definition 1 if we do not know the prior distribution p_X of X. In practice, however, a randomization operator must be chosen before p_X is learned.

It turns out that there exists a sufficient test that has neither of these shortcomings, and there are practically useful randomization operators that satisfy this test. The test is based on comparing the operator's transitional probabilities $p[x \to y]$ for the same $y \in V_Y$ but different $x \in V_X$. Intuitively, if all of the x-values are reasonably likely to be randomized into a given y, then revealing "$R(x) = y$" does not tell too much about x. We call this approach *amplification* because it limits how much some $p[x \to y]$'s can be amplified with respect to others.

4.1 General Approach

Let us define our privacy preserving restriction on randomization operators, and then prove a statement on bounding privacy breaches:

Definition 3. *A randomization operator $R(x)$ is* at most γ-amplifying *for $y \in V_Y$ if*

$$\forall\, x_1, x_2 \in V_X: \ \frac{p[x_1 \to y]}{p[x_2 \to y]} \leqslant \gamma; \tag{2}$$

here $\gamma \geqslant 1$ and $\exists x : p[x \to y] > 0$. Operator $R(x)$ is at most γ-amplifying *if it is at most γ-amplifying for all suitable $y \in V_Y$.*

Statement 1. *Let R be a randomization operator, let $y \in V_Y$ be a randomized value such that $\exists x : p[x \to y] > 0$, and let $0 < \rho_1 < \rho_2 < 1$ be two probabilities from Definition 2. Suppose that R is at most γ-amplifying for y. Revealing "$R(X) = y$" will cause neither upward ρ_1-to-ρ_2 privacy breach nor downward ρ_2-to-ρ_1 privacy breach with respect to any property if the following condition is satisfied:*

$$\frac{\rho_2}{\rho_1} \cdot \frac{1 - \rho_1}{1 - \rho_2} > \gamma. \tag{3}$$

Proof. Note that $\forall x \in V_X$ we have $p[x \to y] > 0$ because otherwise γ is infinite. Let us denote $Y \equiv R(X)$ as a random variable. Consider any distribution p_X; since it is nonzero on at least one $x \in V_X$, we have

$$\mathbf{P}[Y = y] \geqslant \mathbf{P}[X = x] \cdot p[x \to y] > 0.$$

By way of contradiction, let us assume that for property $Q(x)$ we have a ρ_1-to-ρ_2 privacy breach. $Q(x)$ cannot be true for all $x \in V_X$ because $\mathbf{P}[Q(X)] \leqslant \rho_1 < 1$ by the definition of privacy breach. Analogously, $Q(x)$ cannot be false for all $x \in V_X$ because $\mathbf{P}[Q(X) \mid Y = y] \geqslant \rho_2 > 0$. So, the following definitions make sense:

$$x_1 \in \{x \in V_X \mid Q(x) \text{ and } p[x \to y] = \max_{Q(x')} p[x' \to y]\}$$
$$x_2 \in \{x \in V_X \mid \neg Q(x) \text{ and } p[x \to y] = \min_{\neg Q(x')} p[x' \to y]\}$$

In words, x_1 is a private value that has property $Q(x)$ and is most likely to get randomized into y, and x_2 is another value that does not satisfy $Q(x)$ and is least likely to get randomized into y. By the definition of conditional probability,

$$\mathbf{P}[Q(X) \mid Y = y] = \sum_{Q(x)} \mathbf{P}[X = x \mid Y = y] =$$
$$= \sum_{Q(x)} \frac{\mathbf{P}[X = x] \cdot p[x \to y]}{\mathbf{P}[Y = y]} \leqslant$$
$$\leqslant \frac{p[x_1 \to y]}{\mathbf{P}[Y = y]} \cdot \sum_{Q(x)} \mathbf{P}[X = x] = p[x_1 \to y] \cdot \frac{\mathbf{P}[Q(X)]}{\mathbf{P}[Y = y]}$$

and, in the same way,

$$\mathbf{P}[\neg Q(X) \mid Y = y] = \sum_{\neg Q(x)} \mathbf{P}[X = x \mid Y = y] =$$
$$= \sum_{\neg Q(x)} \frac{\mathbf{P}[X = x] \cdot p[x \to y]}{\mathbf{P}[Y = y]} \geqslant$$
$$\geqslant \frac{p[x_2 \to y]}{\mathbf{P}[Y = y]} \cdot \sum_{\neg Q(x)} \mathbf{P}[X = x] = p[x_2 \to y] \cdot \frac{\mathbf{P}[\neg Q(X)]}{\mathbf{P}[Y = y]}$$

We know that $\mathbf{P}\,[Q(X) \mid Y = y] \geqslant \rho_2 > 0$, and it follows from the above that $\mathbf{P}\,[Q(X)] > 0$. Therefore, we can divide the lower inequality by the upper one:

$$\frac{\mathbf{P}\,[\neg Q(X) \mid Y = y]}{\mathbf{P}\,[Q(X) \mid Y = y]} \;\geqslant\; \frac{p\,[x_2 \to y]}{p\,[x_1 \to y]} \cdot \frac{\mathbf{P}\,[\neg Q(X)]}{\mathbf{P}\,[Q(X)]}$$

Let us remember that $R(x)$ is at most γ-amplifying for y:

$$\frac{1 - \mathbf{P}\,[Q(X) \mid Y = y]}{\mathbf{P}\,[Q(X) \mid Y = y]} \;\geqslant\; \frac{1}{\gamma} \cdot \frac{1 - \mathbf{P}\,[Q(X)]}{\mathbf{P}\,[Q(X)]}$$

It remains to notice that

$$\frac{1 - \rho_2}{\rho_2} \geqslant \frac{1 - \mathbf{P}\,[Q(X) \mid Y = y]}{\mathbf{P}\,[Q(X) \mid Y = y]};\quad \frac{1 - \mathbf{P}\,[Q(X)]}{\mathbf{P}\,[Q(X)]} \geqslant \frac{1 - \rho_1}{\rho_1}$$

and we arrive to contradiction with condition (3).

To prove the statement for downward ρ_2-to-ρ_1 breaches, we first represent them as upward ρ'_1-to-ρ'_2 breaches with $\rho'_1 = 1 - \rho_2$ and $\rho'_2 = 1 - \rho_1$, and then note that condition (3) stays satisfied:

$$\frac{\rho'_2}{\rho'_1} \cdot \frac{1 - \rho'_1}{1 - \rho'_2} \;=\; \frac{1 - \rho_1}{1 - \rho_2} \cdot \frac{\rho_2}{\rho_1} \;>\; \gamma.$$

□

We sometimes call inequality (2) *amplification condition* for a given $y \in V_Y$. We need to enforce this condition for all $y \in V_Y$ if we do not want privacy breaches regardless of the randomized private value $R(x)$.

In Example 1, randomization operator R_3 satisfies the amplification condition (2) with $\gamma < 6$. Indeed, for this operator, transitional probabilities are

$$p\,[x \to y] \;=\; \begin{cases} \frac{1}{2}\left(\frac{1}{201} + \frac{1}{1001}\right), & \text{if } y \in [x - 100, x + 100] \\ \frac{1}{2}\left(0 + \frac{1}{1001}\right), & \text{otherwise.} \end{cases}$$

Their fractional difference is $1 + 1001/201 < 6$. Using Statement 1, we can claim that there are no ρ_1-to-ρ_2 upward breaches from $\rho_1 = 1/7 \approx 14\%$ to $\rho_2 = 1/2 = 50\%$, nor the corresponding downward breaches. And we do not even need to know p_X to claim this.

Background Knowledge. Amplification condition (2) limits privacy breaches in the presence of certain kinds of background information about the clients. Suppose that client C_i has private information x_i, and the server knows the value of some function $f(x_i)$, or more generally, an instance of some random variable Z that depends on x_i. From the server's point of view, the probability distribution of the possible values for x_i (i. e. of random variable X), prior and posterior, becomes conditional:

- Prior: $\mathbf{P}\,[X = x] \;\to\; \mathbf{P}\,[X = x \mid Z = z]$
- Posterior: $\mathbf{P}\,[X = x \mid R(X) = y] \;\to$
  $\to\; \mathbf{P}\,[X = x \mid R(X) = y,\, Z = z]$

If the background information is independent from the randomization operator, all transitional probabilities $p\,[x \to y]$ remain the same, so amplification condition remains unaffected and Statement 1 still applies. However, Definition 1 of ρ_1-to-ρ_2 privacy breach in the presence of background knowledge is modified: the breach now occurs when

$$\mathbf{P}\,[Q(X) \mid Z = z] \leqslant \rho_1 \;\text{ and }\; \mathbf{P}\,[Q(X) \mid Y = y,\, Z = z] \geqslant \rho_2.$$

4.2 Itemset Randomization

Now we are going to show how to construct randomization operators that satisfy amplification condition (2) for a given γ and still allow for aggregate data mining by the server. This will be done for one important special case, previously discussed in [9, 17]: randomization of itemsets in association rule mining. Let us start with defining the problem.

Let $\mathcal{I}$ be a set of items, for example products in an on-line store. Suppose there are N clients, each having a *transaction* t_i, where t_i is a subset of $\mathcal{I}$. The items in t_i may represent purchases or preferences of client i. We assume that all transactions have the same size m and that each transaction is an independent instance of some distribution that is not hidden. In real life, transactions have different sizes, but the server can group together transactions according to their nonrandomized size if the size is not hidden.

The server wants to learn itemsets $A \subset \mathcal{I}$ that occur frequently within transactions. That is, it needs all itemsets whose support

$$\text{sup}(A) \;:=\; \frac{\#\{i \mid i = 1 \dots N,\, A \subseteq t_i\}}{N}$$

is at least some minimal support $s_{\min}$. However, the clients are not willing to disclose their personal transactions, so they use randomization. Here we are going to consider the class of randomizations called "select-a-size," defined in [9]. The definition is as follows:

Definition 4. *The* select-a-size *randomization operator has parameters $0 \leqslant \rho \leqslant 1$ and $\{p[j]\}_{j=0}^{m}$, the latter being a probability distribution over $\{0, 1, \dots, m\}$. Given a transaction t, the operator generates another transaction $t' = R(t)$ in three steps:*

1. *The operator selects an integer j at random from the set $\{0, 1, \dots, m\}$ so that $\mathbf{P}\,[j$ is chosen$] = p[j]$;*
2. *It selects j items from t, uniformly at random (without replacement). These items, and no other items of t, are placed into t';*
3. *It considers each item $a \notin t$ in turn and tosses a coin with probability ρ of "heads" and $1 - \rho$ of "tails". All those items for which the coin faces "heads" are added to t'.*

Let us constrain the select-a-size operator with our amplification condition, to ensure the desired limitation on privacy breaches. We shall use the non-strict form (2), because it will allow us to solve an optimization problem. Denote $t' = R(t)$, $m' = |t'|$, $j = |t \cap t'|$, and $n = |\mathcal{I}|$. Then the transitional probabilities of the select-a-size can be written as

$$p\,[t \to t'] \;=\; \frac{p[j]}{\binom{m}{j}} \cdot \rho^{m' - j}(1 - \rho)^{n - m - m' + j}.$$

If there are two transactions t_1 and t_2 with $j_1 = |t_1 \cap t'|$ and $j_2 = |t_2 \cap t'|$, we have

$$\frac{p\,[t_1 \to t']}{p\,[t_2 \to t']} \;=\; \frac{\left(\dfrac{p[j_1]}{\binom{m}{j_1} \cdot \rho^{j_1}(1 - \rho)^{m - j_1}}\right)}{\left(\dfrac{p[j_2]}{\binom{m}{j_2} \cdot \rho^{j_2}(1 - \rho)^{m - j_2}}\right)}.$$

We can call

$$p^{\text{def}}[j] \;:=\; \binom{m}{j} \cdot \rho^{j}(1 - \rho)^{m - j}$$

the "default" probability of selecting j items from t, and "balance" the $p[j]$'s by dividing them by the "default" probabilities:

$$b[j] \;:=\; p[j] \;/\; p^{\text{def}}[j] \tag{4}$$

Then condition (2) becomes

$$\forall j_1, j_2 : \quad b[j_1] \;/\; b[j_2] \;\leqslant\; \gamma. \tag{5}$$

While satisfying this condition, we want to transmit as much aggregate information as possible. Randomized transactions are used by the server in order to determine frequent itemsets. So, we would like to ensure that frequent itemsets in randomized transactions have supports as different as possible from infrequent itemsets, with respect to the standard deviation of the supports. Among the parameters of select-a-size, ρ determines the amount of new items added, and $\{p[j]\}_{j=0}^{m}$ determines the amount of original items deleted. Given ρ, a reasonable heuristic is to set the $p[j]$'s so that, on average, as many original items as possible make it to the randomized transaction. Thus, we are maximizing the following expectation:

$$\nu(p) \;:=\; \mathop{\mathbf{E}}_{t' \sim R(t)} |t \cap t'| \;=\; \sum_{j=0}^{m} j \cdot p[j]$$

Statement 2. *For any nonconstant function $f(j)$ increasing on $j = 0 \ldots m$, the quantity*

$$\nu_f(p) \;:=\; \sum_{j=0}^{m} f(j) \cdot p[j] \;=\; \sum_{j=0}^{m} b[j] \cdot f(j)\, p^{\mathrm{def}}[j]$$

reaches maximum (conditioned by (5) and by $\{p[j]\}_{j=0}^{m}$ being a probability distribution) when, for some $j_* \in \{0, 1, \ldots, m-1\}$, we have

$$\gamma \cdot b[0] = \gamma \cdot b[1] = \ldots = \gamma \cdot b[j_*] = \\ = b[j_* + 1] = \ldots = b[m]. \quad (6)$$

The proof of this statement is in Appendix A.1.

If, instead of trying to have more *items* of t in t', we are trying to have more *k-itemsets* of t in t', then we are maximizing

$$\mathop{\mathbf{E}}_{t' \sim R(t)} \#\{A \subseteq t \mid A \subseteq t', |A| = k\} \;=\; \sum_{j=0}^{m} \binom{j}{k} \cdot p[j],$$

which is also subject to Statement 2 since function $f(j) = \binom{j}{k}$ is increasing. So, the solution again has the form (6), possibly with a different j_*.

Our heuristic thus reduces the problem of selecting parameters ρ and $\{p[j]\}_{j=0}^{m}$ to the problem of selecting ρ and j_*, where j_* is discrete. How to set these two parameters depends on the expected properties of the data, such as how many items are in the itemsets we are mining and what supports these itemsets and their subsets are likely to have. We can use methods from [9] to evaluate the variance in the support estimators, with extra caution when inverting the transition matrix for partial supports since it may be singular for some ρ and j_*.

We computed how much is recoverable after a select-a-size randomization whose parameters are restricted by the amplification condition. The graphs presented here are similar to those in [9]. Again, we use the notion of the *lowest discoverable support* (LDS), which is the lowest possible support that, when recovered after randomization, has a statistically significant separation from zero. By "statistically significant" we mean a separation from zero by four standard deviations. We have computed LDS, in percent, for 1-item, 2-item, and 3-itemsets while varying three numbers:

1. The privacy breach level ρ_1 (in percent), which we define as the least prior probability for an allowed ρ_1-to-ρ_2 privacy breach with $\rho_2 = 50\%$;
2. The transaction size;
3. The number of transactions used for support recovery.

The amplification parameter γ is computed according to formula (3) of Statement 1:

$$\gamma \;=\; \frac{\rho_2}{\rho_1} \cdot \frac{1 - \rho_1}{1 - \rho_2} \;=\; \frac{0.5}{\rho_1} \cdot \frac{1 - \rho_1}{0.5} \;=\; \frac{1}{\rho_1} - 1.$$

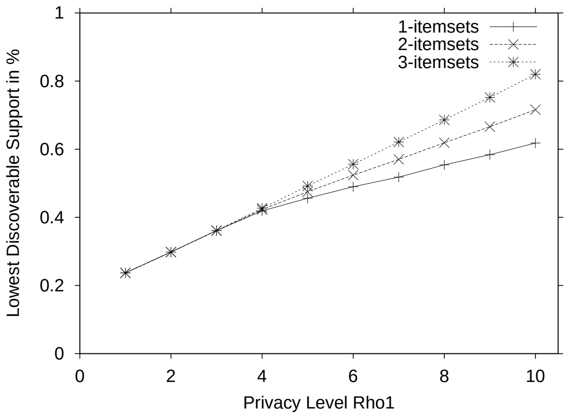

Figure 1: Lowest discoverable support versus breach level ρ_1. 5 million transactions, transaction size is 5.

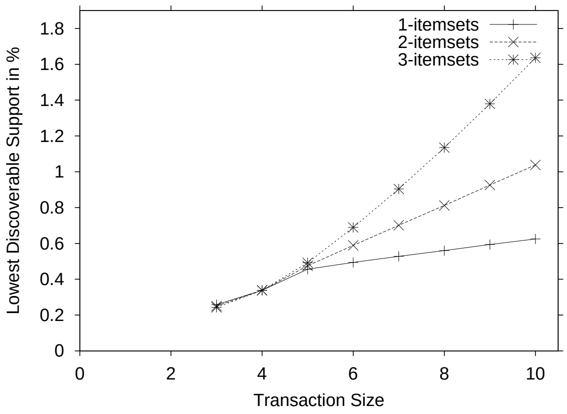

Figure 2: Lowest discoverable support versus transaction size. 5 million transactions, breach level is $\rho_1 = 5\%$.

Parameters j_* and ρ are chosen to minimize the maximum of 1-item, 2-item, and 3-item LDS.

Figure 1 shows how LDS depends on the privacy requirement. We require that there are no breaches with the prior below ρ_1 and posterior at 50%, where $\rho_1 = 1\% \ldots 10\%$. As we see, we can recover supports of about 0.5% when the worst breaches (to 50%) allowed are from 5% to 50%.

The graph on Fig. 2 has its ρ_1 fixed at 5%, but varies transaction size from 3 to 10. Of course, the longer the transaction, the harder it is to recover supports, since there is more private data to be randomized. Finally, the graph on Fig. 3 shows how the number of transactions affects the recovery (in other graphs the default is 5 million transactions). LDS is roughly inversely proportional to the square root of the number of transactions.

5. COMPRESSING RANDOMIZED TRANSACTIONS

When applying select-a-size randomization operator (Definition 4) to transactions, we generate randomized transactions with lots of false items. In fact, the size of each randomized transaction is comparable to the overall number of considered items, which may be in many thousands. Sending these randomized transactions may take significant network resources, and such a database will require a lot of memory. Fortunately, there is a way to compress randomized transactions without causing privacy breaches. The idea is

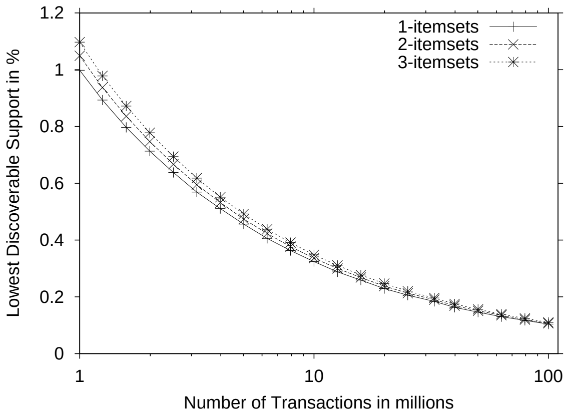

Figure 3: Lowest discoverable support versus number of transactions. Transaction size is 5, breach level is $\rho_1 = 5\%$.

to use a pseudorandom number generator and drop the requirement of treating each item independently.

Definition 5. *A* $(\mathrm{Seed}, n, q, \rho)$-pseudorandom generator *is a function*

$$G:\ \mathrm{Seed} \times \{1, \ldots, n\} \to \{0, 1\}$$

that has the following properties:

1. *$\forall i = 1 \ldots n:\ \mathbf{P}\,[G(\xi, i) = 1 \mid \xi \in_r \mathrm{Seed}] \ = \ \rho$;*
2. *For all integers $1 \leqslant i_1 < i_2 < \ldots < i_q \leqslant n$ and when $\xi \in_r$ Seed, the random variables $G(\xi, i_1)$, $G(\xi, i_2)$, …, $G(\xi, i_q)$ are statistically independent.*

Here Seed *is a finite set, n and q are positive integers, $0 < \rho < 1$, and the sign "$\in_r$" means "is chosen uniformly at random from."*

Let n be the overall number of possible items. Instead of representing a randomized transaction by the list of items it contains, we are going to represent it by a seed $\xi \in$ Seed. Then, for every item number i, we can check whether or not this item belongs to the randomized transaction by computing $G(\xi, i)$: item i belongs to the transaction if and only if $G(\xi, i) = 1$. In other words, there is a mapping τ from seeds to transactions:

$$\tau(\xi) \ = \ \{\text{item } i \mid G(\xi, i) = 1\}. \tag{7}$$

The set Seed in many cases can be the set of Boolean strings $\{0, 1\}^k$, where $k \ll n$.

Suppose we want to randomize transaction t that has m items. We shall define a randomization operator (called *pseudorandom select-a-size*) that uses a pseudorandom generator. The operator is similar to select-a-size operator from Definition 4 and has the same parameters: $0 < \rho < 1$ and $\{p[j]\}_{j=0}^m$, the latter being a probability distribution over $\{0, 1, \ldots, m\}$. Given a transaction t and a $(\mathrm{Seed}, n, q, \rho)$-pseudorandom generator with $q \geqslant m$, the operator generates the seed $\xi = R'(t)$ in three steps:

1. The operator selects an integer j at random from the set $\{0, 1, \ldots, m\}$ so that $\mathbf{P}\,[j \text{ is chosen}] = p[j]$;
2. It selects j items from t, uniformly at random (without replacement). Without loss of generality, assume that items $t[1], t[2], \ldots, t[j]$ are selected.
3. It selects a random seed $\xi \in$ Seed such that

$$G(\xi, t[1]) = \ldots = G(\xi, t[j]) = 1 \ \text{ and}$$
$$G(\xi, t[j+1]) = \ldots = G(\xi, t[m]) = 0.$$

Any seed that satisfies this condition must have equal chances to be selected. This seed is returned as (the seed for) the randomized transaction.

Pseudorandom select-a-size operator will always find some seed at Step 3 because, if we take $\xi \in_r$ Seed, then by Definition 5 the variables $G(\xi, t[1])$, $G(\xi, t[2])$, …, $G(\xi, t[m])$ are statistically independent and therefore can take any combination of values. Moreover, the following statement shows that a transaction randomized by pseudorandom select-a-size operator R' has the same distribution with respect to any *small* subset of items as when it is randomized by the "usual" select-a-size R from Definition 4:

Statement 3. *Let t be a transaction, $|t| = m$, and let $G(\xi, i)$ be a $(\mathrm{Seed}, n, q, \rho)$-pseudorandom generator with $q > m$. Let $R(t)$ be the "usual" select-a-size operator (Definition 4) with parameters ρ and $\{p[j]\}_{j=0}^m$, and let $R'(t)$ be pseudorandom select-a-size operator with the same parameters, with G as its pseudorandom generator. For any itemset A of size at most $q - m$ items and for any $B \subseteq A$ we have (see (7) for the definition of τ):*

$$\mathbf{P}\,[R(t) \cap A = B] \ = \ \mathbf{P}\,[\tau(R'(t)) \cap A = B]. \tag{8}$$

Proof. Let us pay attention only to the items within set $A \cup t$. There are at most q such items. By the definition of pseudorandom generator (Definition 5), as long as seed ξ is chosen uniformly at random, the values of $G(\xi, i)$ for $i \in A \cup t$ are independent and equal 1 with probability ρ. The first two steps of both randomization operators are the same: we select which subset $t_0 \subseteq t$ is going to belong to the randomized transaction. During the last step,

- In the "usual" operator, each item from $A \setminus t$, independently, has probability ρ to get into $R(t)$;
- In the pseudorandom operator, we select a seed $\xi \in_r$ Seed such that $\tau(R'(t)) \cap t = t_0$. Since $\xi \in_r$ Seed, the distribution of items from $A \setminus t$ is not affected by the choice of t_0; each item, independently, has probability ρ to get into $\tau(R'(t))$.

So, both operators induce identical distributions on items within $A \setminus t$, and in particular, satisfy (8). □

It follows from Statement 3 that all the mathematical apparatus for support and variance estimation from [9] is applicable to pseudorandom select-a-size operators as well, as long as we are working with itemsets of size at most $q - m$. Indeed, pseudorandom operator $R(t)$ is a per-transaction operator (it randomizes each transaction independently and its distribution is defined by t). Generally speaking, it is not item-invariant; however, for an itemset A of size at most $q - m$ we have

$$\mathbf{P}\,[\,|\tau(R'(t)) \cap A| = l'\,] = \sum_{B \subseteq A,\ |B| = l'} \mathbf{P}\,[\tau(R'(t)) \cap A = B] =$$
$$= \sum_{B \subseteq A,\ |B| = l'} \mathbf{P}\,[R(t) \cap A = B] \ = \ \mathbf{P}\,[\,|R(t) \cap A| = l'\,] \ = \ p\,[l \to l']$$

where $p\,[l \to l']$ is defined in [9] as

$$p\,[l \to l'] \ = \ \mathbf{P}\,[\,|R(t) \cap A| = l' \ \mid \ |t \cap A| = l\,]$$

and is shown to depend only on l, l', m, $|A|$, and on the parameters of select-a-size randomization. Therefore, we can "bypass" item-invariance.

Now let us find out when pseudorandom select-a-size operator protects from privacy breaches. Here we can no longer restrict ourselves to a few items only, since all items at once are involved in a privacy breach. Instead, we can use the amplification condition (2) and Statement 1 in the same way as we used them for the "usual" select-a-size operator in Section 4.2. The following statement shows that the amplification condition in pseudorandom case translates into exactly the same condition (5) on the randomization parameters:

Statement 4. *Let $R(t)$ and $R'(t)$ be the "usual" and the pseudorandom select-a-size operators respectively, with the same randomization parameters; suppose that R' uses a* $(\text{Seed}, n, q, \rho)$*-pseudorandom generator with $q \geqslant m = |t|$. Then*

$$\forall t_1, t_2, \xi : \ \frac{\mathbf{P}\left[R'(t_1) = \xi\right]}{\mathbf{P}\left[R'(t_2) = \xi\right]} \ = \ \frac{\mathbf{P}\left[R(t_1) = \tau(\xi)\right]}{\mathbf{P}\left[R(t_2) = \tau(\xi)\right]}.$$

Proof. Consider any seed $\xi \in \text{Seed}$ and any transactions t_1 and t_2 of size m. Suppose $\tau(\xi) \cap t_1 = t_1^0$ and $\tau(\xi) \cap t_2 = t_2^0$, and let $j_i = |t_i^0|$ for $j = 1, 2$. Then

$$\begin{aligned}
&\mathbf{P}\left[R'(t_i) = \xi\right] \ = \\
&= \ \mathbf{P}\left[R'(t_i) = \xi \ \mid \ \tau(R'(t_i)) \cap t_i = t_i^0\right] \cdot \mathbf{P}\left[t_i^0 \text{ chosen at Step 2}\right] \\
&= \ \mathbf{P}\left[\xi_r = \xi \ \mid \ \xi_r \in_r \text{Seed}, \ \tau(\xi_r) \cap t_i = t_i^0\right] \cdot p[j_i] \binom{m}{j_i}^{-1} \\
&= \ \frac{\mathbf{P}\left[\xi_r = \xi \ \mid \ \xi_r \in_r \text{Seed}\right]}{\mathbf{P}\left[\tau(\xi_r) \cap t_i = t_i^0 \ \mid \ \xi_r \in_r \text{Seed}\right]} \cdot p[j_i] \binom{m}{j_i}^{-1} \\
&= \ \frac{|\text{Seed}|^{-1}}{\rho^{j_i}(1-\rho)^{m-j_i}} \cdot p[j_i] \binom{m}{j_i}^{-1}
\end{aligned}$$

For the "usual" operator $R(t)$, this probability is (for $t' = \tau(\xi)$, $|t'| = m'$):

$$\begin{aligned}
&\mathbf{P}\left[R(t_i) = t'\right] \ = \ p[j_i] \binom{m}{j_i}^{-1} \cdot \rho^{m'-j_i}(1-\rho)^{n-m-m'+j_i} \\
&= \frac{\rho^{m'}(1-\rho)^{n-m'}}{\rho^{j_i}(1-\rho)^{m-j_i}} \cdot p[j_i] \binom{m}{j_i}^{-1}
\end{aligned}$$

If we divide two probabilities like this, the constant multiplier will cancel out:

$$\frac{\mathbf{P}\left[R'(t_1) = \xi\right]}{\mathbf{P}\left[R'(t_2) = \xi\right]} \ = \ \frac{\mathbf{P}\left[R(t_1) = \tau(\xi)\right]}{\mathbf{P}\left[R(t_2) = \tau(\xi)\right]} \ = \ \frac{b[j_1]}{b[j_2]}$$

where $b[j_i]$ were defined in (4). □

As a consequence of Statement 4, all methodology described in Section 4.2 for select-a-size randomization operators can be applied for pseudorandom select-a-size operators.

It remains to construct an example of a $(\text{Seed}, n, q, \rho)$-pseudorandom generator. For $\rho = 1/2$, these generators, also known as *orthogonal arrays* [11], can be constructed using linear error-correcting codes [14, 16]. A binary linear *error-correcting code* of size n and distance d is the kernel $\{x \in \mathbf{Z}_2^n \mid Mx = \vec{\mathbf{0}}\}$ of a $(k \times n)$-matrix M (called the *parity check matrix*) over the field $\mathbf{Z}_2$ of residues modulo 2 such that any nonzero n-dimensional vector x from the kernel has at least d nonzero coordinates.

The following statement is well-known:

Statement 5. *In a parity check $(k \times n)$-matrix for an error-correcting code of distance d any collection of $d-1$ columns is linearly independent over $\mathbf{Z}_2$. If a vector ξ is chosen uniformly at random from $\mathbf{Z}_2^k$, then in $M^T\xi$ any collection of $q = d-1$ bits is distributed as q independent random bits, each with probability $1/2$ of being zero.*

The proof of this statement is in Appendix A.2.

Let n be the number of all possible items, let m be the original transaction size (considered fixed), and let ρ be the default probability of an item in the select-a-size operator, represented in the form $\rho = a/2^b$ where a and b are integers. Suppose the server is interested in supports of itemsets of size up to s, but no more; then we need a $(\text{Seed}, n, q, \rho)$-pseudorandom generator with $q = m + s$. Consider an error-correcting code with size bn and distance $d = bq + 1$; let M be its parity check matrix with k rows, and let $\text{Seed} = \{0, 1\}^k$.

Given $\xi \in \{0, 1\}^k$ and $i \in \{1, \ldots, n\}$, our pseudorandom generator computes a bit as follows:

1. Compute vector $x = M^T\xi$ over $\mathbf{Z}_2$;
2. Take the following subvector of size b bits:
$$x_i \ = \ \langle x[b(i-1)], \ x[b(i-1)+1], \ \ldots, \ x[b\,i-1]\rangle$$
3. Output 1 if and only if
$$\sum_{j=0}^{b-1} x\left[b(i-1)+j\right] \cdot 2^j \ < \ a.$$

This pseudorandom generator satisfies Definition 5. Indeed, by Statement 5, if $\xi \in_r \{0, 1\}^k$ then any combination of bq bits of $x = M^T\xi$ is independently distributed, each bit being 1 with probability 1/2. As a consequence, any combination of q disjoint b-bit subvectors is independently distributed, and each b-bit subvector is "showing" a binary representation of a number below a with probability $a/2^b = \rho$.

How well can we compress randomized transactions using error-correcting codes? Consider, for example, the Bose-Chaudhuri-Hocquenghem (BCH) codes [14, 16]; there, for any positive integers r and $l \leqslant 2^{r-1} - 1$, we have a parity check matrix of size $rl \times (2^r - 1)$ with distance $2l + 1$. If we are dealing with transactions of size $m = 10$ and are interested in itemsets of size up to $s = 5$ and if $\rho = 1/2$ making $b = 1$, for example, then we need distance $b(m+s)+1 = 16$, which makes $l = 8$. If there are $100,000$ items overall, we need $r = 17$, and hence the size of the compressed transaction is $rl = 136$ bits, much less than the ordinary way which needs $100,000$ bits. For $\rho = 1/16$ we have $b = 4$ and $b(m+s)+1 = 61$ making $l = 30$, $r = 19$, and compressed transaction becomes 570 bits, while ordinary way needs at least $H(1/16) \cdot 10^5 > 33,729$ bits.

6. WORST-CASE INFORMATION

Amplification approach from Section 4 is designed to be independent on the prior distribution, to depend only on the randomization operator itself. There can be other ways to restrict disclosure, other privacy measures that depend both on the prior distribution of private data and on the operator. In this section we consider a class of privacy measures inspired by Shannon's information theory [19], adjusted so that they bound privacy breaches.

In the paper [1] the authors introduce a measure of privacy which is a function of mutual information between two distributions, the original data distribution and the randomized data distribution. Suppose that X is a random variable such that each data record is its independent instance. Let $Y = R(X)$ be another random variable (R is randomization) such that each randomized data record is an instance of Y. Then mutual information $I(X;Y)$ is

$$\begin{aligned}
I(X;Y) :=\ & KL\left(p_{X,Y} \ \|\ p_X p_Y\right) = \\
=\ & \mathop{\mathbf{E}}_{y \sim Y} KL\left(p_{X|Y=y} \ \|\ p_X\right)
\end{aligned}$$

where $KL\,(p_1 \,\|\, p_2)$ is Kullback-Leibler distance between the distributions $p_1(x)$ and $p_2(x)$ of two random variables:

$$
\begin{aligned}
KL\,(p_1 \,\|\, p_2) &:= \mathop{\mathbf{E}}_{x \sim p_1} \log \frac{p_1(x)}{p_2(x)},\\
p_{X,Y}(x,y) &:= \mathbf{P}\,[X = x, Y = y],\\
p_{X\,|\,Y=y}(x) &:= \mathbf{P}\,[X = x \mid Y = y].
\end{aligned}
$$

It is assumed that the larger $I(X;Y)$ is, the less privacy is preserved. Unfortunately, there are situations where privacy is obviously not preserved, but mutual information does not show any sign of trouble. Here is an example:

Example 2. Let our private data be just one bit: $V_X = \{0, 1\}$. Assume that both 0 and 1 are equally likely: $\mathbf{P}\,[X = 0] = \mathbf{P}\,[X = 1] = 1/2$. Now consider two randomizations, $Y_1 = R_1(X)$ and $Y_2 = R_2(X)$. The first randomization, given $x \in V_X$, outputs x with probability 60% and outputs $1 - x$ with probability 40%:

$$
\begin{aligned}
\mathbf{P}\,[Y_1 = x \mid X = x] &= 0.6,\\
\mathbf{P}\,[Y_1 = 1 - x \mid X = x] &= 0.4
\end{aligned}
$$

The second randomization R_2 can output 0, 1, or "empty record" e. Whatever its input x is, it outputs e with probability 99.99%, otherwise it outputs x with probability 0.0099% and $1 - x$ with probability 0.0001%:

$$
\begin{aligned}
\mathbf{P}\,[Y_2 = e \mid X = x] &= 0.9999,\\
\mathbf{P}\,[Y_2 = x \mid X = x] &= 0.000099 = 99 \cdot 10^{-6},\\
\mathbf{P}\,[Y_2 = 1 - x \mid X = x] &= 0.000001 = 1 \cdot 10^{-6}.
\end{aligned}
$$

Intuitively, R_2 is a very poor randomizer since if we see, say, $Y_2 = 1$, then we know with very high probability that $X = 1$:

$$
\begin{aligned}
&\mathbf{P}\,[X = 1 \mid Y_2 = 1] \;=\; \mathbf{P}\,[X = 0 \mid Y_2 = 0] \;=\\
&\quad = \;\frac{99 \cdot 10^{-6} \cdot 0.5}{99 \cdot 10^{-6} \cdot 0.5 + 1 \cdot 10^{-6} \cdot 0.5} \;=\; 0.99
\end{aligned}
$$

For Y_1, this probability is only 0.6, which is much more reasonable. What does mutual information indicate, however? Let us compute $KL\,(p_{X\,|\,Y_i=y} \,\|\, p_X)$ for $i = 1, 2$ and $y = 0, 1, e$:

$$
\begin{aligned}
&y = 0, 1: \;\log \frac{\mathbf{P}\,[X = y \mid Y_1 = y]}{\mathbf{P}\,[X = y]} = \log \frac{0.6}{0.5} \approx 0.2630,\\
&\log \frac{\mathbf{P}\,[X = 1 - y \mid Y_1 = y]}{\mathbf{P}\,[X = 1 - y]} = \log \frac{0.4}{0.5} \approx -0.3219,\\
&KL\,(p_{X\,|\,Y_1=y} \,\|\, p_X) \approx\\
&\approx 0.6 \cdot 0.2630 - 0.4 \cdot 0.3219 \approx 0.02905
\end{aligned}
$$

$$
\begin{aligned}
&y = 0, 1: \;\log \frac{\mathbf{P}\,[X = y \mid Y_2 = y]}{\mathbf{P}\,[X = y]} = \log \frac{0.99}{0.5} \approx 0.9855,\\
&\log \frac{\mathbf{P}\,[X = 1 - y \mid Y_2 = y]}{\mathbf{P}\,[X = 1 - y]} = \log \frac{0.01}{0.5} \approx -5.6439,\\
&KL\,(p_{X\,|\,Y_2=y} \,\|\, p_X) \approx\\
&\approx 0.99 \cdot 0.9855 - 0.01 \cdot 5.6439 \approx 0.91921;\\
&y = e,\; x = 0, 1: \;\log \frac{\mathbf{P}\,[X = x \mid Y_2 = e]}{\mathbf{P}\,[X = x]} = \log \frac{0.5}{0.5} = 0,\\
&KL\,(p_{X\,|\,Y_2=e} \,\|\, p_X) = 0
\end{aligned}
$$

Now we can compute and compare mutual informations. For Y_1, both of $KL\,(p_{X\,|\,Y_1=y} \,\|\, p_X)$ for $y = 0, 1$ are the same, so the average is

$$I(X;Y_1) \approx 0.02905;$$

For Y_2, the Kullback-Leibler distances are very different, and since $\mathbf{P}\,[Y_2 = e] = 0.9999$, the average is

$$I(X;Y_2) \approx 0.9999 \cdot 0 + 0.0001 \cdot 0.91921 \ll I(X;Y_1).$$

Thus, counter to intuition, mutual information says that R_2 is more privacy-preserving than R_1.

Of course, mutual information fails to detect privacy breaches in Example 2 because they are very infrequent: they occur only in 0.01% randomizations. But once a breach occurs, it is detectable, and noone wants to be the unfortunate client who has the breach. Mutual information averages all Kullback-Leibler distances; however, by looking at these distances without taking the average, some breaches become visible. Indeed, in Example 2, distances $KL\,(p_{X\,|\,Y_1=y} \,\|\, p_X)$ for R_1 are both small ($\approx$ 0.02905), whereas for R_2 some distances are big, e.g.

$$KL\,(p_{X\,|\,Y_2=1} \,\|\, p_X) \;\approx\; 0.91921$$

This indicates that revealing "$Y_2 = 1$" may lead to a privacy breach. The measure that shows the worst possible Kullback-Leibler distance rather than averages them will do better at measuring privacy. We come to the following definition:

Definition 6. *Let X and Y be discrete random variables. We define* worst-case information *as follows:*

$$I_w(X;Y) \;:=\; \max_y \, KL\,(p_{X\,|\,Y=y} \,\|\, p_X).$$

Instead of the logarithm, we can use a different numerical function $f(t)$ as long as $t\,f(t)$ is a convex function on the interval $t > 0$:

Definition 7. *Let X and Y be discrete random variables, and let $f(t)$ be a numerical function such that $t\,f(t)$ is convex on $t > 0$. We define worst-case information* with respect to f *as follows:*

$$
\begin{aligned}
I_w^f(X;Y) \;&:=\; \max_y \, KL^f\,(p_{X\,|\,Y=y} \,\|\, p_X), \quad \text{where}\\
KL^f\,(p_1 \,\|\, p_2) \;&:=\; \mathop{\mathbf{E}}_{x \sim p_1} \, f\big(p_1(x)/p_2(x)\big).
\end{aligned}
$$

Now we are going to show that knowing worst-case information gives a bound on upward privacy breaches.

Statement 6. *Suppose that revealing $R(X) = y$ for some y causes an upward ρ_1-to-ρ_2 privacy breach with respect to property $Q(X)$. Then*

$$\rho_2 \cdot f\left(\frac{\rho_2}{\rho_1}\right) \;+\; (1 - \rho_2) \cdot f\left(\frac{1 - \rho_2}{1 - \rho_1}\right) \;\leqslant\; I_w^f(X; R(X)).$$

The proof of this statement is in Appendix A.3.

As claimed in Statement 6, worst-case information allows us to bound upward privacy breaches. But what to do with downward privacy breaches? It turns out that they are bounded by a measure similar to worst-case information, but in a way "inside-out," or inverse worst-case information. Here is the definition:

Definition 8. *Let X and Y be discrete random variables, and let $f(t)$ be a numerical function such that $t\,f(t)$ is convex on $t > 0$. We define* inverse worst-case information *with respect to f as follows:*

$$J_w^f(X;Y) \;:=\; \max_y \, KL^f\,(p_X \,\|\, p_{X\,|\,Y=y}).$$

Even though Kullback-Leibler distance is called "distance," it is not symmetrical, so usually $J_w^f(X;Y) \neq I_w^f(X;Y)$. The main difference between the two is that the value of $I_w^f(X;Y)$ depends on the behavior of properties likely after Y has been revealed,

Measure	R_1	R_2	R_3
$I(X;R(X))$	1.27	2.32	0.55
$I_w(X;R(X))$	3.90	2.33	0.55
$J_w(X;R(X))$	1.72	∞	0.49

Table 2: The values of average-case and worst-case information measures in Example 1.

whereas the value of $J_w^f(X;Y)$ depends on the behavior of properties likely *a priori*. Indeed, in $I_w^f(X;Y)$, the average is taken with respect to distribution $p_{X|Y=y}$, while in $J_w^f(X;Y)$ the average is with respect to p_X. The inverse worst-case information is related to downward breaches in the same way as the straight worst-case information to upward breaches. Let us formulate it in the following statement.

Statement 7. *Suppose that revealing $R(X) = y$ for some y causes a downward ρ_2-to-ρ_1 privacy breach with respect to property $Q(X)$. Then*

$$\rho_2 \cdot f\left(\frac{\rho_2}{\rho_1}\right) + (1-\rho_2)\cdot f\left(\frac{1-\rho_2}{1-\rho_1}\right) \leqslant J_w^f(X;R(X)).$$

The proof of this statement is in Appendix A.4.

Table 2 gives average-case and worst-case information measures (with $f = \log$) for the three randomization operators from Example 1 (see Section 3). The table shows that R_1 is more sensitive to upward privacy breaches, R_2 is more sensitive to downward privacy breaches, and R_3 has little sensitivity to both of them. The same trend was shown in Table 1.

7. CONCLUSION

We presented a new defintion of privacy breaches, and developed a general approach, called amplification, that provably limits breaches. Amplification can be used to limit privacy breaches with respect to any single-record property. More importantly, unlike earlier approaches, this approach does not require knowledge of the data distribution to provide privacy guarantees. We instantiated this approach for the problem of mining association rules, and derived the amplification condition for the select-a-size randomization operator.

Next, we gave a method for compressing long randomized transactions by using pseudorandom generators, and showed that this could reduce their sizes by orders of magnitude. Finally, we defined several new information-theoretical privacy measures that provably bound privacy breaches.

We conclude with some interesting directions for future research.

- How do we extend amplification to continuous distributions?
- What is the relationship between the specific randomization operators, and the tradeoff between privacy and accuracy? In particular, how do we identify the randomization operator and parameters that will provide the highest accuracy in the mining model for a given level of privacy breaches?
- Are there ways to combine the randomization and the secure multi-party computation approaches that work better than either approach alone?

8. REFERENCES

[1] D. Agrawal and C. C. Aggarwal. On the design and quantification of privacy preserving data mining algorithms. In *Proceedings of the 20th Symposium on Principles of Database Systems*, Santa Barbara, California, USA, May 2001.

[2] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In *Proceedings of the 20th International Conference on Very Large Databases*, Santiago, Chile, September 1994.

[3] R. Agrawal and R. Srikant. Privacy preserving data mining. In *ACM SIGMOD Conference on Management of Data*, pages 439–450, Dallas, Texas, May 2000.

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

[5] C. Clifton and D. Marks. Security and privacy implications of data mining. In *ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery*, pages 15–19, May 1996.

[6] The Economist. *The End of Privacy*, May 1999.

[7] V. Estivill-Castro and L. Brankovic. Data swapping: Balancing privacy against precision in mining for logic rules. In M. Mohania and A. Tjoa, editors, *Data Warehousing and Knowledge Discovery DaWaK-99*, pages 389–398. Springer-Verlag Lecture Notes in Computer Science 1676, 1999.

[8] European Union. *Directive on Privacy Protection*, October 1998.

[9] A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke. Privacy preserving mining of association rules. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery in Databases and Data Mining*, pages 217–228, Edmonton, Alberta, Canada, July 23–26 2002.

[10] A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke. Privacy preserving mining of association rules. In *Proc. of the 8th ACM SIGKDD Int'l Conference on Knowledge Discovery and Data Mining*, Edmonton, Canada, July 2002.

[11] A. S. Hedayat, N. J. A. Sloane, and J. Stufken. *Orthogonal Arrays: Theory and Applications.* Springer Verlag, August 1999. 440 pp.

[12] M. Kantarcioglu and C. Clifton. Privacy-preserving distributed mining of association rules on horizontally partitioned data. In *ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery*, June 2002.

[13] Y. Lindell and B. Pinkas. Privacy preserving data mining. In *CRYPTO*, pages 36–54, 2000.

[14] F. J. C. MacWilliams and N. J. A. Sloane. *The Theory of Error-Correcting Codes.* North-Holland, Amsterdam, 1978. 762 pp.

[15] Office of the Information and Privacy Commissioner, Ontario. *Data Mining: Staking a Claim on Your Privacy*, January 1998.

[16] O. Pretzel. *Error-Correcting Codes and Finite Fields.* Oxford University Press, 1992. 398 pp.

[17] S. J. Rizvi and J. R. Haritsa. Maintaining data privacy in association rule mining. In *Proceedings of the 28th International Conference on Very Large Data Bases*, Hong Kong, China, August 2002.

[18] S. J. Rizvi and J. R. Haritsa. Privacy-preserving association rule mining. In *Proc. of the 28th Int'l Conference on Very Large Databases*, August 2002.

[19] C. E. Shannon. Communication theory of secrecy systems. *Bell System Technical Journal*, 28-4:656–715, 1949.

[20] K. Thearling. Data mining and privacy: A conflict in making. *DS**, March 1998.

[21] J. Vaidya and C. W. Clifton. Privacy preserving association rule mining in vertically partitioned data. In *Proc. of the 8th ACM SIGKDD Int'l Conference on Knowledge Discovery and Data Mining*, Edmonton, Canada, July 2002.

APPENDIX

A. PROOFS

A.1 Proof of Statement 2

Proof. Let us show that, for any $\{p[j]\}_{j=0}^m$ that satisfies (5), we can construct a distribution of type (6) for which the value of $\nu_f(p)$ is at least as large. The idea is to raise and lower some $p[j]$'s while keeping the other $p[i]$'s, $i \neq j$, in constant relation to each other, and so that $\nu_f(p)$ does not decrease in the process.

Given $j = 0, 1, \ldots, m$, suppose we increase $p[j]$ by the factor of y and decrease all other $p[i]$'s, for $i \neq j$, by a factor of x (thereby producing distribution $\{\tilde{p}[j]\}_{j=0}^m$). Then

$$\sum_{i=0}^{m} \tilde{p}[i] = y \cdot p[j] + x \cdot \sum_{i \neq j} p[i] = 1.$$

We can see that $y = y(x)$ is a linear function of x; therefore, the function

$$\nu_f(\tilde{p}) = \sum_{i=0}^{m} f(i) \cdot \tilde{p}[i] = f(j) \cdot y\, p[j] + \sum_{i \neq j} f(i) \cdot x\, p[i]$$

is also linear with respect to x. We have $\nu_f(\tilde{p}) = \nu_f(p)$ when $x = 1$ and $\nu_f(\tilde{p}) = f(j)$ when $x = 0$ (because then $\tilde{p}[j] = 1$ and $\tilde{p}[i] = 0$ for $i \neq j$). So,

$$\nu_f(\tilde{p}) = f(j) + x \cdot (\nu(p) - f(j)).$$

We say that we *raise* probability $p[j]$ if we change the distribution by decreasing x (starting from $x = 1$) and increasing y until $\tilde{b}[j] = \gamma \cdot \tilde{b}[i]$ for some number $i \neq j$. We can raise only those $p[j]$'s for which $f(j) > \nu(p)$, so that $\nu_f(\tilde{p})$ can only increase. This raising always stops before $\tilde{p}[j]$ reaches 1 or (and) any other probability reaches 0. Analogously, we say that we *lower* $p[j]$ if we increase x and decrease y until $\tilde{b}[j] = \tilde{b}[i]/\gamma$ for some $i \neq j$. We can lower only if $f(j) \leqslant \nu_f(p)$ to prevent $\nu_f(\tilde{p})$ from decreasing. Note that raising and lowering does not affect relations $\tilde{b}[i_1]/\tilde{b}[i_2]$ for $i_1 \neq j \neq i_2$.

We modify the distribution in two steps. First, we lower $p[0]$ and raise $p[m]$. We have $b[m] = \gamma \cdot b[0]$: indeed, when we lowered $p[0]$, $b[0]$ became the smallest of all $b[j]$'s, so it will set the limit to raising $p[m]$. Then, we repeat the following process: choose $p[j]$ that can be lowered or raised, and lower it or raise. Clearly, neither $p[0]$ nor $p[m]$ will ever be chosen since they limit each other and since always $f(0) < \nu(p) < f(m)$. A lowered $b[j]$ becomes equal to $b[0]$, a raised one becomes equal to $b[m]$. Any probability can be lowered only once and will never be raised again because once $f(j) \leqslant \nu(p)$, it stays this way. Any probability can be raised only once and then possibly lowered. After no more than $2(m-1)$ raisings and lowerings, there is nothing to change. For $j_* = \max\{j \mid f(j) \leqslant \nu(p)\}$, the form (6) is attained. □

A.2 Proof of Statement 5

Proof. Suppose, w.l.o.g., that columns $M_1, M_2, \ldots, M_{d-1}$ are not linearly independent. Then there is a nonzero linear combination that equals a zero vector:

$$x_1 M_1 + x_2 M_2 + \ldots + x_{d-1} M_{d-1} = \vec{0}.$$

Extend x to an n-dimensional vector by setting coordinates $x_d, \ldots, x_n$ to zeros. We get a nonzero vector from the matrix's kernel, which has less than d nonzero coordinates — a contradiction.

Consider any $d-1$ coordinates in $M^T \xi$; w.l.o.g., assume they are the first $d-1$ coordinates. Since the first $d-1$ rows of M^T are linearly independent, they form a $(d-1) \times k$-submatrix M' of rank $d-1$. For any $(d-1)$-dimensional vector v, equation $M'\xi = v$ has the same number 2^{k-d+1} of solutions. When every vector ξ is equally likely, every vector $v = M'\xi$ is therefore also equally likely, that is, its coordinates behave independently. □

A.3 Proof of Statement 6

We first prove two simple lemmas and a corollary.

Lemma 1. *If function* $t\, f(t)$ *is convex (or strictly convex) on the interval* $t > 0$, *then so is function* $f(1/t)$.

Proof. Let $0 < t_1 < t_2$ be two different numbers, and let $0 < \lambda < 1$. For strict convexity, we need

$$\lambda f\left(\frac{1}{t_1}\right) + (1-\lambda) f\left(\frac{1}{t_2}\right) > f\left(\frac{1}{\lambda t_1 + (1-\lambda) t_2}\right); \quad (9)$$

for non-strict convexity, just replace ">" with "⩾." Denote

$$t = \lambda t_1 + (1-\lambda) t_2, \quad \alpha = \frac{\lambda t_1}{t};$$

it is clear that $0 < \alpha < 1$. Then, by strict convexity of $t\, f(t)$ we have

$$\alpha \frac{1}{t_1} f\left(\frac{1}{t_1}\right) + (1-\alpha) \frac{1}{t_2} f\left(\frac{1}{t_2}\right) > \\ > \left(\alpha \frac{1}{t_1} + (1-\alpha) \frac{1}{t_2}\right) f\left(\alpha \frac{1}{t_1} + (1-\alpha) \frac{1}{t_2}\right).$$

Substitution of the definition of α gives

$$\frac{\lambda}{t} f\left(\frac{1}{t_1}\right) + (1-\lambda) \frac{1}{t} f\left(\frac{1}{t_2}\right) > \\ > \left(\lambda \frac{1}{t} + (1-\lambda) \frac{1}{t}\right) f\left(\lambda \frac{1}{t} + (1-\lambda) \frac{1}{t}\right),$$

which is equivalent to (9). □

Lemma 2. *Let* X, Y, *and* Z *be discrete random variables such that* Z *is independent from* Y *given* X, *and let* $t\, f(t)$ *be convex on* $t > 0$. *Then, for all possible* y,

$$\begin{aligned} KL^f\left(p_{Z|Y=y} \parallel p_Z\right) &\leqslant KL^f\left(p_{X|Y=y} \parallel p_X\right) \quad \text{and} \\ KL^f\left(p_Z \parallel p_{Z|Y=y}\right) &\leqslant KL^f\left(p_X \parallel p_{X|Y=y}\right). \end{aligned}$$

Proof. Let us prove the first and then the second inequality using the definition of KL^f. We shall use Jensen's inequality $\mathbf{E}\ g(\tau) \geqslant g(\mathbf{E}\ \tau)$ with respect to function $g(t) = f(1/t)$, which is convex on $t > 0$ by Lemma 1.

$$KL^f\left(p_{X|Y=y} \parallel p_X\right) = \mathop{\mathbf{E}}_{x \sim X|Y=y} f\left(\frac{\mathbf{P}[X = x \mid Y = y]}{\mathbf{P}[X = x]}\right)$$

$$= \mathop{\mathbf{E}}_{z \sim Z|Y=y} \mathop{\mathbf{E}}_{x \sim X|\substack{Z=z \\ Y=y}} f\left(1 \Big/ \frac{\mathbf{P}[X = x]}{\mathbf{P}[X = x \mid Y = y]}\right) \geqslant$$

$$\geqslant \mathop{\mathbf{E}}_{z \sim Z|Y=y} f\left(1 \Bigg/ \left(\mathop{\mathbf{E}}_{x \sim X|\substack{Z=z \\ Y=y}} \frac{\mathbf{P}[X = x]}{\mathbf{P}[X = x \mid Y = y]}\right)\right);$$

Using the independence of Z from Y given X, we transform the internal expectation to the desired fraction:

$$\begin{aligned} &\mathop{\mathbf{E}}_{x \sim X \,|\, {Z=z \atop Y=y}} \frac{\mathbf{P}[X=x]}{\mathbf{P}[X=x \mid Y=y]} \;=\\ &=\; \mathop{\mathbf{E}}_{x \sim X} \frac{\mathbf{P}[X=x \mid Z=z, Y=y]}{\mathbf{P}[X=x \mid Y=y]}\\ &=\; \mathop{\mathbf{E}}_{x \sim X} \frac{\mathbf{P}[Z=z \mid X=x, Y=y]}{\mathbf{P}[Z=z \mid Y=y]}\\ &=\; \mathop{\mathbf{E}}_{x \sim X} \frac{\mathbf{P}[Z=z \mid X=x]}{\mathbf{P}[Z=z \mid Y=y]} \;=\; \frac{\mathbf{P}[Z=z]}{\mathbf{P}[Z=z \mid Y=y]}. \end{aligned}$$

The first inequality is thus proven. The second inequality is very analogous:

$$\begin{aligned} &KL^f\left(p_X \parallel p_{X|Y=y}\right) \;=\; \mathop{\mathbf{E}}_{x \sim X} f\left(\frac{\mathbf{P}[X=x]}{\mathbf{P}[X=x \mid Y=y]}\right)\\ &=\; \mathop{\mathbf{E}}_{z \sim Z} \mathop{\mathbf{E}}_{x \sim X|Z=z} f\left(1 \Big/ \frac{\mathbf{P}[X=x \mid Y=y]}{\mathbf{P}[X=x]}\right) \;\geqslant\\ &\geqslant\; \mathop{\mathbf{E}}_{z \sim Z} f\left(1 \Big/ \left(\mathop{\mathbf{E}}_{x \sim X|Z=z} \frac{\mathbf{P}[X=x \mid Y=y]}{\mathbf{P}[X=x]}\right)\right); \end{aligned}$$

$$\begin{aligned} &\mathop{\mathbf{E}}_{x \sim X|Z=z} \frac{\mathbf{P}[X=x \mid Y=y]}{\mathbf{P}[X=x]} \;=\\ &= \mathop{\mathbf{E}}_{x \sim X|Y=y} \frac{\mathbf{P}[X=x | Z=z]}{\mathbf{P}[X=x]} = \mathop{\mathbf{E}}_{x \sim X|Y=y} \frac{\mathbf{P}[Z=z | X=x]}{\mathbf{P}[Z=z]}\\ &= \mathop{\mathbf{E}}_{x \sim X|Y=y} \frac{\mathbf{P}[Z=z | X=x, Y=y]}{\mathbf{P}[Z=z]} = \frac{\mathbf{P}[Z=z | Y=y]}{\mathbf{P}[Z=z]}. \end{aligned}$$

Both inequalities are now proven. □

Corollary 1. *Under the conditions in Lemma 2, we have*

$$I_w^f(Z;Y) \;\leqslant\; I_w^f(X;Y).$$

Proof. Follows immediately from the first inequality of Lemma 2: if for every number in one set there is at least as large number in the other set, then the maximal number of the first set $\leqslant$ the maximal number of the other. □

Proof of Statement 6. Now we have all the tools to prove the bound on upward privacy breaches.

Proof. Let us denote $Y = R(X)$ and

$$P_1 \;=\; \mathbf{P}[Q(X)], \quad P_2 \;=\; \mathbf{P}[Q(X) \mid Y=y].$$

By Definition 2, we have $P_1 \leqslant \rho_1 < \rho_2 \leqslant P_2$. Let us define α, q_1, and q_2 as follows:

$$\begin{aligned} q_1 \;&=\; \rho_2 + \alpha(1-P_2), \quad q_2 \;=\; \rho_1 - \alpha P_1,\\ \alpha \;&=\; \frac{\rho_2 - \rho_1}{P_2 - P_1}. \end{aligned}$$

It is clear that $0 < \alpha \leqslant 1$, and therefore

$$\begin{aligned} 0 \;\leqslant\; \rho_2 \;\leqslant\; q_1 \;\leqslant\; 1 - (P_2 - \rho_2) \;\leqslant\; 1,\\ 0 \;\leqslant\; \rho_1 - P_1 \;\leqslant\; q_2 \;\leqslant\; \rho_1 \;\leqslant\; 1; \end{aligned}$$

so, q_1 and q_2 can serve as probabilities. Let us employ them, then. Define a Boolean random variable Z that depends on X as follows:

1. If $Q(X)$, then Z says “true” with probability q_1;
2. If $\neg Q(X)$, then Z says “true” with probability q_2.

Of course, Z is independent from Y given X, so Corollary 1 is applicable:

$$KL^f\left(p_{Z|Y=y} \parallel p_Z\right) \;\leqslant\; I_w^f(Z;Y) \;\leqslant\; I_w^f(X;Y).$$

It remains to check that this inequality is exactly what we are proving. Indeed, denote $I = I_w^f(X;Y)$ and “open up” the definition of KL^f:

$$\begin{aligned} &\mathbf{P}[Z \mid Y=y] \cdot f\left(\frac{\mathbf{P}[Z \mid Y=y]}{\mathbf{P}[Z]}\right) \;+\\ &\qquad +\; \mathbf{P}[\neg Z \mid Y=y] \cdot f\left(\frac{\mathbf{P}[\neg Z \mid Y=y]}{\mathbf{P}[\neg Z]}\right) \;\leqslant\; I. \end{aligned}$$

Now compute the prior and posterior probabilities of Z:

$$\begin{aligned} \mathbf{P}[Z] \;&=\; q_1 \cdot P_1 \;+\; q_2 \cdot (1-P_1) \;=\\ &=\; P_1\left(\rho_2 + \alpha(1-P_2)\right) \;+\; (1-P_1)(\rho_1 - \alpha P_1)\\ &=\; \rho_1 \;+\; P_1\left(\rho_2 - \rho_1\right) \;-\; \alpha\, P_1\left(P_2 - P_1\right)\\ &=\; \rho_1 \;+\; P_1\left(\rho_2 - \rho_1\right) \;-\; P_1\left(\rho_2 - \rho_1\right) \;=\; \rho_1; \end{aligned}$$

analogously,

$$\begin{aligned} \mathbf{P}[Z \mid Y=y] \;&=\; q_1 \cdot P_2 \;+\; q_2 \cdot (1-P_2) \;=\\ &=\; P_2\left(\rho_2 + \alpha(1-P_2)\right) \;+\; (1-P_2)(\rho_1 - \alpha P_1)\\ &=\; \rho_1 \;+\; P_2\left(\rho_2 - \rho_1\right) \;+\; \alpha\left(1-P_2\right)(P_2 - P_1)\\ &=\; \rho_1 \;+\; P_2\left(\rho_2 - \rho_1\right) \;+\; (1-P_2)(\rho_2 - \rho_1) \;=\; \rho_2. \end{aligned}$$

The statement is proven. □

A.4 Proof of Statement 7

Let us start with another corollary of Lemma 2.

Corollary 2. *Under the conditions in Lemma 2, we have*

$$J_w^f(Z;Y) \;\leqslant\; J_w^f(X;Y).$$

Proof. Follows from the second inequality of Lemma 2 in the same way as Corollary 1 from the first. □

Proof of Statement 7. We are now ready to prove Statement 7.

Proof. The proof is almost analogous to that of Statement 6. We only have to change places between prior and posterior distributions. Namely, we define

$$\begin{aligned} P_2 \;&=\; \mathbf{P}[Q(X)], \quad P_1 \;=\; \mathbf{P}[Q(X) \mid Y=y],\\ q_1 \;&=\; \rho_2 + \alpha(1-P_2), \quad q_2 \;=\; \rho_1 - \alpha P_1,\\ \alpha \;&=\; \frac{\rho_2 - \rho_1}{P_2 - P_1}. \end{aligned}$$

Again, by Definition 2 we have $P_1 \leqslant \rho_1 < \rho_2 \leqslant P_2$; we define Z exactly like before, and in the end get

$$\begin{aligned} &\mathbf{P}[Z] \cdot f\left(\frac{\mathbf{P}[Z]}{\mathbf{P}[Z \mid Y=y]}\right) \;+\\ &\qquad +\; \mathbf{P}[\neg Z] \cdot f\left(\frac{\mathbf{P}[\neg Z]}{\mathbf{P}[\neg Z \mid Y=y]}\right) \;\leqslant\; J_w^f(X;Y), \end{aligned}$$

where

$$\mathbf{P}[Z \mid Y=y] \;=\; \rho_1, \quad \mathbf{P}[Z] = \rho_2.$$

The statement is proven. □