Utility-Driven Anonymization in Data Publishing*

Mingqiang Xue⋆ Panagiotis Karras◊ Chedy Raïssi§ Hung Keng Pung⋆

⋆National University of Singapore ◊Rutgers University §INRIA, Nancy Grand-Est

ABSTRACT

Privacy-preserving data publication has been studied intensely in the past years. To date, all existing approaches transform data values by random perturbation or generalization. In this paper, we introduce a radically different data anonymization methodology. Our proposal aims to maintain a certain amount of *patterns*, defined in terms of a set of properties of interest that hold for the original data. Such properties are represented as linear relationships among data points. We present an algorithm that generates a set of anonymized data that strictly preserves these properties, thus maintaining specified *patterns* in the data. Extensive experiments with real and synthetic data show that our algorithm is efficient, and produces anonymized data that affords high utility in several data analysis tasks while safeguarding privacy.

Categories and Subject Descriptors

H.2.7 [**Database Management**]: Database Administration—*Security, integrity, and protection*; H.2.8 [**Database Management**]: Database Applications—*Data Mining*; K.4.1 [**Computers and Society**]: Public Policy Issues—*Privacy*

General Terms

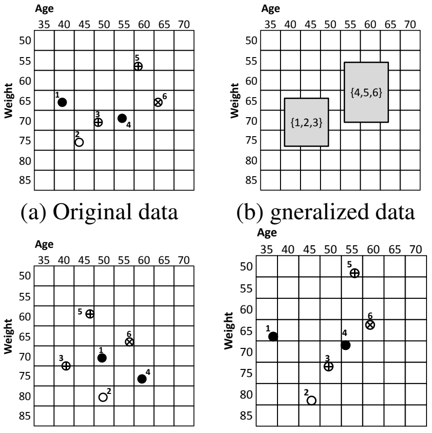
Algorithms, Security, Experimentation

1. INTRODUCTION

Organizations often possess data that need to be made public for the common good. Yet such data may contain sensitive personal information. Several methodologies for data anonymization have been proposed, with the aim of protecting the privacy of persons involved while maintaining as much of the utility of the published data as possible. Existing approaches for data anonymization transform the data by either *generalizing* or *perturbing* data values. Generalization-based approaches [10, 9, 2] group records into *equivalence classes* (ECs), and render the records within the same EC indistinguishable by *generalizing* their values on some pre-selected quasi-identifying attributes (QIs) to the same range(s).

id	*age*	*weight*	*disease*
1	42	66	Gastritis ●
2	40	76	Diabetes ○
3	49	73	Pneumonia ⊕
4	54	68	Gastritis ●
5	55	53	Pneumonia ⊕
6	60	66	Alzheimer ⊗

Table 1: Sample medical micro-data



(a) Original data (b) gneralized data

(c) Perturbed data (d) Our approach

Figure 1: Comparison of anonymization paradigms

Table 1 shows a sample of medical microdata records. *Age* and *Weight* are *quasi-identifying attributes* [12]; knowledge of those attributes' values allows an adversary to re-identify the person involved. *Disease* is a *sensitive attribute*; it contains information that entails a privacy risk for the persons concerned. Figure 1(a) visualizes these microdata in the two-dimensional space formed by the two quasi-identifiers [5], *Age*×*Weight*. An *anonymization* of these data by the generalization-based k-anonymity model with $k = 3$ could form two ECs out of them, one containing records $\{1, 2, 3\}$ and one out of $\{4, 5, 6\}$. Figure 1(b) presents these two ECs as rectangular regions in the two-dimensional QI-space. All records within the same generalized region become indistinguishable as far as their QIs are concerned. Following the paradigm of random perturbation [1], an attribute value is modified by adding to the original value a random variable uniformly or normally distributed in a predefined interval $[-\alpha, +\alpha]$. This perturbation effectively contains an adversary's capacity to re-identify the record of a specific person. In the running example, Figure 1(c) presents an example of how the published data may look after random perturbation.

A careful examination of Figures 1(b) and (c) reveals that, though the anonymized data obtained from generalization and perturbation achieves privacy objectives, important topological relationships between the data points are lost. In generalization, topological in-

*Work supported by Singapore's MOE AcRF grant T1 251RES0807.

CIKM'11, October 24–28, 2011, Glasgow, Scotland, UK.

formation is completely obscured within the generalized regions; in perturbation, topological information is destroyed due to blind modification. However, topological information is essential in data mining tasks such as ranking, clustering or skyline queries. Hence, the data obtained from generalization or perturbation is not suitable for these tasks. Popular generalization-based approaches [12, 10, 8, 9, 2] and random perturbation-based approaches [1, 4, 11, 3, 14, 6] all inherit such shortcomings.

In this paper, we propose a novel data anonymization paradigm that addresses the above drawbacks. Given a microdata table $\mathcal{T}$, we define a process that specifies certain *properties of interest* (PoIs) among the QI attributes of $\mathcal{T}$. The set of PoIs describes the data characteristics that the anonymization should maintain. Each PoI is expressed as a linear relationship between a subset of QI attribute values. We develop a scheme to obtain an anonymized table $\mathcal{T}^A$ that satisfies all defined PoIs. $\mathcal{T}^A$ is nearly randomly and uniformly sampled the space for all possible data that satisfy the PoIs, so as to afford privacy for the original data. With our approach, we get a result as in Figure1(d) for the sample medical data. Notably, the data pattern in (d) preserves the original pattern in (a) much more faithfully than that in (b) and (c). Still, the data values in (d) are modified in a way that preserves privacy, and they do not appear exactly the same as the original data.

2. NOTATIONS AND DEFINITIONS

Let $\mathcal{T}$ be a table with n data records and m QI attributes. Entry $t_{i,j}$ refers to the data value in the i^{th} row and j^{th} column in $\mathcal{T}$. We first focus on describing the scheme for anonymzing 1D data, and later generalize it to work for a table with multiple QIs. Let $D = \{d_1, \ldots, d_n\}$ be a particular 1D data vector (i.e., a QI column of $\mathcal{T}$) that is subject to anonymization and $X = \{x_1, \ldots, x_n\}$ be the corresponding set of variables that express the anonymized form of D. Then we define a *property of interest* as follows.

DEFINITION 1. *A property of interest (PoI) on data vector D is a linear relationship of the form* $\sum_i^{d_i \in D} c_i d_i \leq \lambda$ *among values in D, where* c_i *is the coefficient of* d_i *and* $\lambda \in \mathbb{R}^+$ *is a user-defined constant.*

Our scheme operates in two phases, the *properties extraction phase* and the *value substitution phase*, to find a set of new data $D^A = \{d_1^A, \ldots, d_n^A\}$ for the variables in X that satisfy $\mathcal{Q}$.

3. PROPERTIES EXTRACTION PHASE

Our overall proposal does not constraint the types of linear relationships that may be defined as PoIs. It is up to the data vendor and legitimate data recipients to decide what form of PoIs are important. However, in order to make our proposal concrete and illustrate the *properties extraction phase*, we introduce a particular type of PoIs that we use in the rest of this paper, *locality*.

DEFINITION 2 (LOCALITY). *The* locality *of data values* d_i *and* d_k *with respect to data value* d_j*, denoted as* $loc_{d_j}(d_i, d_k)$ *is a linear relationship of the form* $|d_i - d_j| \odot |d_j - d_k|$*, where* $\odot \in \{\geq, \leq\}$ *makes the relationship true.*

Locality captures relative distance information of two data values with respect to a third data value. The distance between d_i and d_j is denoted as $d_{i,j}$. Without loss of generality, we assume that $d_i < d_k$. A locality property is most informative when d_j lies between d_i and d_k, i.e. $d_i \leq d_j \leq d_k$. Otherwise, it suffices to know whether $d_j < d_i$ or $d_j > d_k$ to deduce the property that holds, independently of the value of d_j.

3.1 Extraction of Localities

Without loss of generality, we assume that D is sorted in non-decreasing order. At first glance, each combination of i, j and k can form a *locality*; thus, the total number of *locality* properties is $\binom{n}{3}$. A naive *locality* extraction algorithm would have to enumerate all possible combinations of i, j, k in $O(n^3)$. Still, some of the *localities* generated by such a process would be *redundant*. For example, from $d_{1,3} \leq d_{3,4}$ we can infer $d_{1,3} \leq d_{3,5}$. We utilize the following two rules for pruning redundant localities:

- *Rule 1:* If $d_i < d_k$ and $d_{i,j} \leq d_{j,k}$, then $d_{i,j} \leq d_{j,k'}$, $\forall k' > k$ and $d_{i',j} \leq d_{j,k}, \forall i' \in [i, k]$.
- *Rule 2:* If $d_i < d_k$ and $d_{i,j} \geq d_{j,k}$, then $d_{i,j} \geq d_{j,k'}$, $\forall k' \in [i, k]$ and $d_{i',j} \geq d_{j,k}, \forall i' < i$.

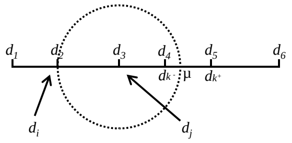


Figure 2: Illustration of locality extraction

Based on the two pruning rules, we propose an algorithm that generates a complete set of *localities*, $\mathcal{P}_{locs}$, whose size is $O(n^2)$. We use the following simple example to illustrate the rationale behind the algorithm. Figure 2 shows a set of data values $D = \{d_1, \ldots, d_6\}$ from which localities are to be extracted. Suppose we wish to retrieve localities for $i = 2$ and $j = 3$. A naive approach would try all possible values of $k > j$ (i.e., $k = 4, 5$ or 6) to determine $loc_{d_3}(d_2, d_4)$, $loc_{d_3}(d_2, d_5)$ and $loc_{d_3}(d_2, d_6)$, respectively. However, we can avoid this enumeration thanks to a simple geometrical observation: A circle centered at d_j with radius $d_j - d_i$, intersecting the D axis at breakpoint μ, implies that for all k values such that $d_k \leq \mu$, it is $d_j - d_i \geq d_k - d_j$. Similarly, for all k values such that $d_k \geq \mu$, $d_j - d_i \leq d_k - d_j$. Let d_{k^-} be the largest value in D less than μ and d_{k^+} be the smallest value in D that is greater than μ. In our example, d_{k^-} is d_4 and d_{k^+} is d_5 (Figure 2). It then suffices to derive the localities $d_j - d_i \geq d_{k^-} - d_j$ and $d_j - d_i \leq d_{k^+} - d_j$, instead of generating one for each possible k. Algorithm 1 shows a pseudocode for our *locality* extraction algorithm. Arguably, a small subset of *localities* $\mathcal{P} \subset \mathcal{P}_{locs}$ can provide a balance between utility and privacy, where $\mathcal{P}$ can be creating by sampling from $\mathcal{P}_{loc}$, or via other selection algorithms.

Algorithm 1: Locality Extraction Algorithm

```
Data: Original data $D$
Result: a set of localities $\mathcal{P}_{locs}$
$\mu, k^-, k^+ \leftarrow 0$;
for $j \leftarrow 2$ to $n$ do
    for $i \leftarrow 1$ to $j - 1$ do
        $\mu \leftarrow 2d_j - d_i$;
        if $\mu \leq d_n$ then
            $j^* \leftarrow \min\{\ell | d_\ell \geq \mu\}$;
            $k^- \leftarrow j^* - 1$;
            $k^+ \leftarrow j^*$;
            $\mathcal{P}_{locs}$.add$(\{i, j, k^-, \geq\})$;
            $\mathcal{P}_{locs}$.add$(\{i, j, k^+, \leq\})$;
        else
            $\mathcal{P}_{locs}$.add$(\{i, j, n, \geq\})$;
return $\mathcal{P}_{locs}$;
```

4. VALUE SUBSTITUTION PHASE

In this section, we tackle the second step of our anonymization scheme, i.e., the value substitution phase. The problem we face is to find a set of values for the variables in X so that all the constraints in $\mathcal{Q}$ are satisfied. In addition, we also aim to find a solution in which the *correlation* between the anonymized and the original data is weak. The value substitution algorithm should ideally give a solution *randomly* and *uniformly* sampled from the solution space.

To achieve our goal, we propose a *Random Walk* algorithm. In this algorithm, D and X are viewed as vectors in a n-dimensional space $\mathbb{R}^n$, i.e. $D = (d_1\ d_2\ \dots\ d_n)^T$ and $X = (x_1\ x_2\ \dots\ x_n)^T$. In addition, we introduce a set of constraints $\mathcal{H}$ that defines the domain values of the attributes: $\mathcal{H} : \gamma_{min} \leq X^T \leq \gamma_{max}$. For instance, we can use $\mathcal{H}$ to ensure *Age* values are bounded between 1 and 120. Trivially, the set of all constraints on X, $\tilde{\mathcal{Q}} = \mathcal{Q} \cup \mathcal{H}$, defines a bounded polyhedron $\mathcal{S}$ in $\mathbb{R}^n$. Any point in $\mathcal{S}$ is a feasible assignment to X that satisfies the constraints $\tilde{\mathcal{Q}}$. Thus, our problem is to randomly select a point in $\mathcal{S}$. Our algorithm exploits the fact that D is an already known solution in $\mathcal{S}$; it initiates a random walk from D and arrives at another internal point within $\mathcal{S}$. We ensure that the random walk always stays within the bounds of $\mathcal{S}$; thus, the arrival point corresponds to an acceptable value assignment to all the variables in X. To minimize the correlation of the destination point to the original data D, the *Random Walk* algorithm operates in an iterative manner. As the number of iterations increases, the probability distribution of the location of the final destination tends to be uniform [13].

We now elaborate on the details of the random walk within $\mathcal{S}$. Each random walk iteration is characterized by two parameters: its direction, ΔX, and the length of walk in the direction, θ. In the following we describe the derivation of ΔX and θ:

- *Walking direction* ΔX. Let $\Delta X = (\Delta x_1\ \Delta x_2 \dots \Delta x_n)$. First, n numbers are randomly chosen from a normal distribution to form a directional vector. Then, the directional vector is normalized to a unit vector and returned.
- *Walking length* θ. The walking length is bounded by a maximum value l in direction ΔX, beyond which the walk leads to a point outside the solution space. The actual walking length θ is then randomly and uniformly chosen from $[0, l]$.

In the following, we derive the maximum walking length l.

4.1 Maximum Walking Length

To calculate l, we convert the linear inequalities in $\mathcal{Q}$ to a set of linear equalities by adding a non-negative slack variable v_i to the left-hand side of each inequality:

$$\textstyle\sum_j^{1\leq j\leq n} c_{i,j} \cdot x_j + v_i = \lambda_i, \text{ where } v_i \geq 0 \tag{1}$$

Let V be the set of all slack variables, i.e. $V = \{v_1, \dots, v_{|\mathcal{Q}|}\}$. (X, V) represents the vector for all variables in the system.

Similar to the walking direction ΔX for the variables in X, we can also introduce a direction vector $\Delta V = (\Delta v_1\ \dots\ \Delta v_{|\mathcal{Q}|})$ for the slack variables V. ΔV. Then $(\Delta X, \Delta V)$ is the direction vector in a particular random walk. In the following, we try to express each Δv_i in terms of ΔX. As the destination of the random walk is within $\mathcal{S}$, the following equation holds after the walk:

$$\textstyle\sum_j^{1\leq j\leq n} c_{i,j}(x_j + \theta \cdot \Delta x_j) + (v_i + \theta \cdot \Delta v_i) = \lambda_i \tag{2}$$

From Equations 1 and 2, we derive:

$$\textstyle\sum_j^{1\leq j\leq n} c_{i,j} \Delta x_j + \Delta v_i = 0 \tag{3}$$

The above equality can be rewritten to express Δv_i as:

$$\Delta v_i = -\textstyle\sum_j^{1\leq j\leq n} c_{i,j} \Delta x_j \tag{4}$$

Since $V^T \geq 0$ is always required and the values in X should always be in their predefined ranges, the following system of inequalities is formed:

$$\begin{cases} V + \theta \cdot \Delta V \geq 0 \\ \gamma_1 \leq X + \theta \cdot \Delta X \leq \gamma_1 \end{cases} \tag{5}$$

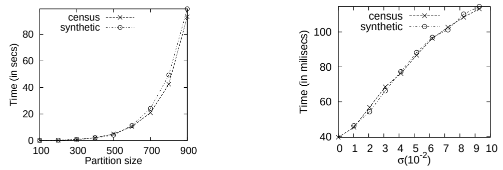
Let $[\theta_{min}, \theta_{max}]$ be the interval that defines the feasible range of θ in the above system; then $l = \max\{0, \theta_{max}\}$.

4.2 Table Anonymization

The anonymization scheme we have developed applies to 1D data vectors. To anonymize a table, we anonymize each QI attribute column independently. Moreover, instead of treating a single column as a single 1D data set, we partition it to segments, and treat each segment independently. We emphasize that this approach does not contribute to the anonymization itself; it is only a mechanism to assist in defining PoIs and processing the data efficiently.

5. EXPERIMENTAL EVALUATION

We now conduct experiments of our anonymization scheme, using both real and synthetic data. Our first data set is a sample of the IPUMS USA census data[1] for the year 2008. It consists of 75K data records; we extract *Age*, *Birth place*, and *Occupation* as QIs. Our second data set is a synthetic one created by the randdataset tool[2]. We create a table with 3 columns and 10K rows, where the columns are independently selected from $[0, 1]$, treated as normalized values of the same four attributes as in the IPUMS data. To compare with ℓ-diversification schemes, we employ the Adult dataset[3]; we extract its first 30K tuples, and treat attributes *Age*, *Final weight* and *Education years* as QIs, and *Occupation* as the $\mathcal{SA}$. All algorithms were developed in Java, and experiments ran on a 3GHz CPU, 2GB RAM machine running Windows XP.



(a) Properties extraction time (b) Value substitution time

Figure 3: Algorithm runtime

5.1 Running Time

We start by measuring the runtime of our algorithm. We increase the partition size from 100 to 900, and measure the average time for extracting all the localities for a partition with Algorithm 1. Figure 3(a) plots our results for both the census and synthetic data. As expected, time grows quadratically in partition size. Still, locality extraction runs in reasonable time even for large partition sizes.

In our next experiment, we fix the partition size to be 100; we run property extraction and randomly sample a number out of the set of all *localities* produced in a partition. We denote the percentage of sampled PoIs as σ. Then, we run our value substitution algorithm with the chosen set of localities as constraints, taking 4000 random-walk iterations. Figure 3(b) shows our results. The runtime grows linearly in the number of PoIs, as our analysis in Section 4 predicts.

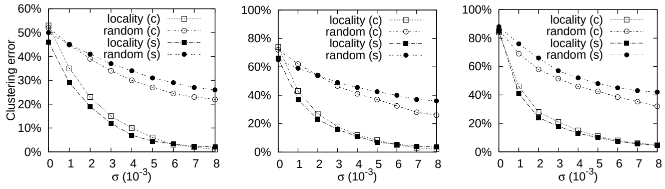
5.2 Clustering Quality

In this experiment, we evaluate our method by conducting a popular data mining operation, k-means clustering, over the anonymized data set. We produce anonymized forms $\mathcal{T}^A$ of the same original data table $\mathcal{T}$ using our approach, and a random perturbation-based scheme [1], ensuring that both of them effect the same amount

[1] http://usa.ipums.org/usa/

[2] http://pgfoundry.org/projects/randdataset

[3] http://archive.ics.uci.edu/ml/datasets/Adult



(a) 2-clustering (b) 4-clustering (c) 6-clustering

Figure 4: Data quality for clustering

of *distortion* on the data. The distortion is assessed as follows:

$$Dst(\mathcal{T}, \mathcal{T}^A) = \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} \left| \frac{t_{i,j} - t_{i,j}^A}{t_{i,j}} \right|}{m \cdot n} \tag{6}$$

This metric measures the average relative error in each entry of the anonymized data with respect to the original data. We first set the size of randomly sampled PoIs σ for our scheme and measure the distortion Dst it incurs; then, we tune the perturbation interval in [1] so that it effects the same (or less) distortion Dst', allowing for a small divergence ϵ. Both compared schemes maintain exact data values, hence their results are amenable to clustering. As grounds of assessment we use the following clustering error metric:

$$CE(\mathcal{T}, \mathcal{T}^A) = \frac{1}{2n} \sum_{i=1}^{k} |C_i(\mathcal{T}) \cup C_i(\mathcal{T}^A)| - |C_i(\mathcal{T}) \cap C_i(\mathcal{T}^A)|$$

where $C_i(\mathcal{T})$ and $C_i(\mathcal{T}^A)$ are the sets of data records in the i^{th} cluster based on the original data $\mathcal{T}$ and the anonymized data $\mathcal{T}^A$, respectively. The clustering error measures the percentage of data records that fail to be grouped in the correct cluster. We measure this error as a function of the size of sampled PoIs σ for our scheme, on which the amount of effected distortion for both methods depends. The partition size is 100, and the random walk makes 4000 iterations. The results are shown in Figure 4, for 3 different k values in k-means clustering, for both the census (c) and synthetic (s) data. Our scheme consistently outperforms the one based on random perturbation.

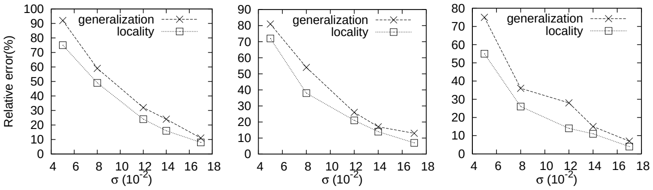
5.3 Answering Aggregate Queries

Next, we study the suitability of using the anonymized data generated with our approach for answering aggregate queries using the Adult dataset. We compare the results derived with our scheme against the generalization-based Mondrian algorithm for ℓ-diversity [7]. We design three types of aggregate queries:

- *Query 1:* SELECT AVG(*Age*) FROM $\mathcal{T}$ WHERE *Fw* $> \tau_{fw}$ AND *Edu* $> \tau_{edu}$ AND (*Occ* = o_1 OR . . . OR *Occ* = o_b)
- *Query 2:* SELECT AVG(*Fw*) FROM $\mathcal{T}$ WHERE *Age* $> \tau_{age}$ AND *Edu* $> \tau_{edu}$ AND (*Occ* = o_1 OR . . . OR *Occ* = o_b)
- *Query 3:* SELECT AVG(*Edu*) FROM $\mathcal{T}$ WHERE *Age* $> \tau_{age}$ AND *Fw* $> \tau_{fw}$ AND (*Occ* = o_1 OR . . . OR *Occ* = o_b)

For each query, the parameters τ_{age}, τ_{fw} and τ_{edu} take the values that are randomly chosen from their attribute domains. The set $\{o_1, \ldots, o_b\}$ is a random subset of all possible occupation values of random size $b \in [1, 14]$. Each query asks for the average value of one QI attribute based on predicates on other attributes.

We first anonymize the Adult dataset using generalization with $\ell = 4, 6, 8, 10$ and 12. We measure the relative errors obtained with generalization with respect to the *distortion* (Equation 6). To measure the distortion of generalized data, we select the mean value of each attribute within the EC as its representative value. Hence, for each version of anonymized dataset $\mathcal{T}^\ell$ under a particular value of ℓ, we can compute a distortion value Dst_ℓ. Then, for each Dst_ℓ value, we tune the amount of PoIs σ used in our approach, until we arrive at an anonymized data set having the same or just a bit more distortion than Dst_ℓ. The partition size is 20, and the number of random walking iterations is $40,000$. Next, we create 2,000 instances for each of the three queries, execute them over these data, and average the accuracy results for each query type. When estimating the answers to range predicates with ℓ-diversified data, we assume that QI values are uniformly distributed within their ECs, and calculate the estimates accordingly. The accuracy of a query answer is assessed by the relative error $\frac{|\phi - \phi^A|}{\phi}$, where ϕ (ϕ_A) is the answer based on the original (anonymized) data.



(a) Query 1 (b) Query 2 (c) Query 3

Figure 5: Answering aggregate queries

Our results are shown in Figure 5. Remarkably, for all three queries, the results with our scheme are more accurate than those by ℓ-diversification, even though both incur the same distortion. This result shows that our method preserves more utility than Mondrian.

6. CONCLUSION

This paper has proposed a simple, yet effective, methodology for data anonymization, which allows a data owner to publish *exact*, instead of generalized, values, yet also preserves patterns among the data. Our scheme extracts a set of properties of interest as linear inequalities, which the anonymized data, generated by a random walk process, preserve. As opposed to traditional *privacy-driven* approaches, our approach is *utility-driven*. Our experiments verify that data anonymized by our approach allows for better or similar performance in data analysis tasks compared to data undergoing the same distortion under other anonymization methods.

7. REFERENCES

[1] R. Agrawal and R. Srikant. Privacy-preserving data mining. *SIGMOD Rec.*, 29(2):439–450, 2000.

[2] J. Cao, P. Karras, P. Kalnis, and K.-L. Tan. SABRE: a Sensitive Attribute Bucketization and REdistribution framework for t-closeness. *The VLDB Journal*, 20(1):59–81, 2011.

[3] K. Chen, G. Sun, and L. Liu. Towards attack-resilient geometric data perturbation. In *SDM*, 2007.

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

[5] G. Ghinita, P. Karras, P. Kalnis, and N. Mamoulis. A framework for efficient data anonymization under privacy and accuracy constraints. *ACM TODS*, 34(2):1–47, 2009.

[6] M. Hay, V. Rastogi, G. Miklau, and D. Suciu. Boosting the accuracy of differentially private histograms through consistency. *PVLDB*, 3(1), 2010.

[7] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Mondrian multidimensional k-anonymity. In *ICDE*, 2006.

[8] N. Li, T. Li, and S. Venkatasubramanian. t-closeness: Privacy beyond k-anonymity and ℓ-diversity. In *ICDE*, 2007.

[9] N. Li, T. Li, and S. Venkatasubramanian. Closeness: A new privacy measure for data publishing. *IEEE TKDE*, 22(7):943–956, 2010.

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

[11] S. Mukherjee, Z. Chen, and A. Gangopadhyay. A privacy-preserving technique for euclidean distance-based mining algorithms using fourier-related transforms. *The VLDB Journal*, 15(4):293–315, 2006.

[12] P. Samarati. Protecting respondents' identities in microdata release. *IEEE TKDE*, 13(6):1010–1027, 2001.

[13] R. L. Smith. Efficient monte carlo procedures for generating points uniformly distributed over bounded regions. *Operations Research*, 32(6):1296–1308, 1984.

[14] Y. Tao, X. Xiao, J. Li, and D. Zhang. On anti-corruption privacy preserving publication. In *ICDE*, pages 725–734, 2008.