Fast Data Anonymization with Low Information Loss

Gabriel Ghinita[1] Panagiotis Karras[2] Panos Kalnis[1] Nikos Mamoulis[2]

[1]Dept. of Computer Science
National University of Singapore
{ghinitag, kalnis}@comp.nus.edu.sg

[2]Dept. of Computer Science
University of Hong Kong
{pkarras, nikos}@cs.hku.hk

ABSTRACT

Recent research studied the problem of publishing microdata without revealing sensitive information, leading to the privacy preserving paradigms of k-anonymity and ℓ-diversity. k-anonymity protects against the identification of an individual's record. ℓ-diversity, in addition, safeguards against the association of an individual with specific sensitive information. However, existing approaches suffer from at least one of the following drawbacks: *(i)* The information loss metrics are counter-intuitive and fail to capture data inaccuracies inflicted for the sake of privacy. *(ii)* ℓ-diversity is solved by techniques developed for the simpler k-anonymity problem, which introduces unnecessary inaccuracies. *(iii)* The anonymization process is inefficient in terms of computation and I/O cost.

In this paper we propose a framework for efficient privacy preservation that addresses these deficiencies. First, we focus on one-dimensional (i.e., single attribute) quasi-identifiers, and study the properties of optimal solutions for k-anonymity and ℓ-diversity, based on meaningful information loss metrics. Guided by these properties, we develop efficient heuristics to solve the one-dimensional problems in linear time. Finally, we generalize our solutions to multi-dimensional quasi-identifiers using space-mapping techniques. Extensive experimental evaluation shows that our techniques clearly outperform the state-of-the-art, in terms of execution time and information loss.

1. INTRODUCTION

Organizations, such as hospitals, need to release microdata (e.g., medical records) for research and other public benefit purposes. However, sensitive personal information (e.g., disease of a specific person) may be revealed in this process. Conventionally, identifying attributes, such as name or social security number, are not disclosed in order to protect privacy. Still, recent research [5, 17] has demonstrated that this is not sufficient, due to the existence of *quasi-identifiers* in the released microdata. Quasi-identifiers are sets of attributes (e.g., ⟨ZIP, Sex, DateOfBirth⟩), which can be joined with information obtained from diverse sources (e.g., public voting registration data) in order to reveal the identity of individual records.

To address this threat, Sweeney and Samarati proposed the k-anonymity model [16, 17]: for every record in a released table there should be at least $k-1$ other records identical to it along a set of *quasi-identifying* attributes. Records with identical quasi-identifier values constitute an *equivalence class*. k-anonymity is commonly achieved by generalization (e.g., show only the area code instead of the exact phone number) or suppression (i.e., hide some values of the quasi-identifier), which inadvertently lead to information loss. Still, the data should remain as informative as possible, in order to be useful in practice. Hence a trade-off between privacy and information loss emerges.

Recently, the concept of ℓ-diversity [13] was introduced to address the limitations of k-anonymity. The latter may disclose sensitive information when there are many identical *sensitive attribute* (SA) values within an equivalence class[1] (e.g., all persons suffer from the same disease). ℓ-diversity prevents uniformity and background knowledge attacks by ensuring that at least ℓ SA values are *well-represented* in each equivalence class (e.g., the probability to associate a tuple with an SA value is bounded by $1/\ell$ [20]). Ref. [13] suggests that any k-anonymity algorithm can be adapted to achieve ℓ-diversity, by altering the equivalence class validation condition. However, such an approach may yield excessive information loss, as we demonstrate in the following example.

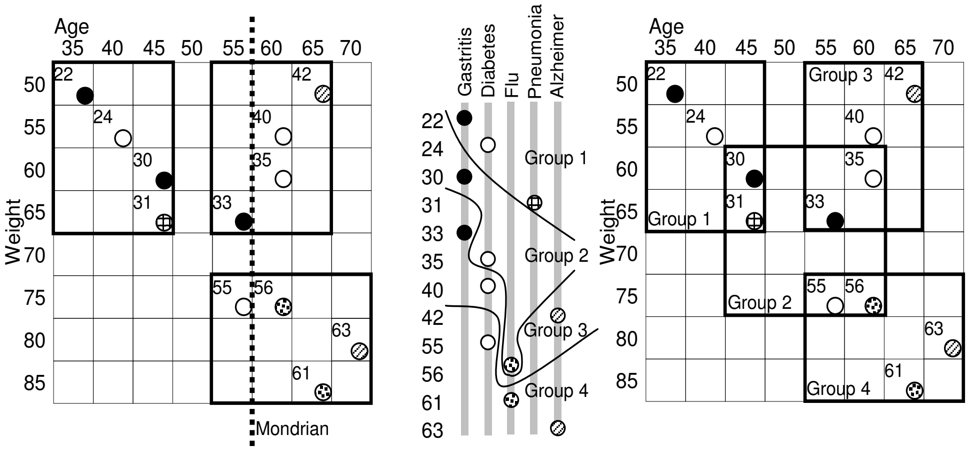
Consider the microdata in Figure 1(a), where the combination of $\langle Age, Weight\rangle$ is the quasi-identifier and $Disease$ is the sensitive attribute. Let the required degree of anonymity be k=4. The current state-of-the-art k-anonymity algorithm (i.e., *Mondrian* [10]) sorts the data points along each dimension (i.e., Age and $Weight$), and partitions across the dimension with the widest normalized range of values. In our example, the normalized ranges for both dimensions are the same. Mondrian selects the first one (i.e., Age) and splits it into segments $35-55$ and $60-70$ (see Figure 1(b)). Further partitioning is not possible, because any split would result in groups with less than 4 records. We propose a different approach. First we map the multi-dimensional quasi-identifier to a 1-D value. In this example we use an 8×8 Hilbert space filling curve (see Section 5 for details); other

[1] k-anonymity remains a useful concept, suitable for cases where the sensitive attribute is implicit or omitted (e.g., a database containing information for convicted persons, regardless of specific crimes).

VLDB '07, September 23-28, 2007, Vienna, Austria.
Copyright 2007 VLDB Endowment, ACM 978-1-59593-649-3/07/09.

Age	Weight	Disease		1D
35	50	Gastritis	●	22
40	55	Diabetes	○	24
45	60	Gastritis	●	30
45	65	Pneumonia	⊕	31
55	65	Gastritis	●	33
60	60	Diabetes	○	35
60	55	Diabetes	○	40
65	50	Alzheimer	▨	42
55	75	Diabetes	○	55
60	75	Flu	⊛	56
65	85	Flu	⊛	61
70	80	Alzheimer	▨	63

(a) Original microdata



(b) k-anonymous groups (c) 1-D l-diversity (d) l-diverse groups

Figure 1: k-anonymity (k=4) and ℓ-diversity (ℓ=3) examples

mappings are also possible. The resulting sorted 1-D values are shown in Figure 1(a) (column *1D*). Next, we partition the 1-D space. We prove that the optimal 1-D partitions are non-overlapping and contain between k and $2k-1$ records. We obtain 3 groups which correspond to 1-D ranges [22..31], [33..42] and [55..63]. The resulting 2-D partitions are enclosed by three rectangles in Figure 1(b). Note that our method causes less information loss. For instance, there is a 1/12 probability for a person who weighs 65kg and is 45 years old, to suffer from pneumonia (since there are 12 dataspace cells covered by the group with Age $35-45$ and $Weight$ $50-65$). In contrast, according to Mondrian, the probability is only 1/40. Clearly, our partitioning is more accurate.

The advantages of our approach are more prominent when applied to ℓ-diversity. This problem is more difficult because, in order to cover a variety of SA values, the optimal 1-D partitioning may include overlapping ranges. For example, if ℓ=3, the second group in Figure 1(c) contains tuples {30, 35, 56}, whereas the third group contains tuples {33, 40, 42}. Nevertheless, we prove that there exist optimal partitionings consisting of only consecutive ranges with respect to each individual value of the sensitive attribute. Based on this property, we develop a heuristic which essentially groups together records that are close to each other in the 1-D space, but have different sensitive attribute values. The four resulting groups are shown in Figure 1(d). From the result we can infer, for instance, that no person younger than 55 suffers from Alzheimer's. On the other hand, if we use Mondrian, we cannot partition the space at all, because any possible disjoint partitioning would violate the ℓ-diversity property. For example, if the Age axis was split into segments $35-55$ and $60-70$ (i.e., same as the k-anonymity case), then gastritis would appear in the left-side partition with probability 3/6, which is larger than the allowed 1/3. Since all tuples are included in the same partition, according to Mondrian, young or old persons have the same probability to suffer from Alzheimer's. Obviously the resulting information loss is unacceptable.

The previous example demonstrates that k-anonymity techniques, such as Mondrian, are not appropriate for the ℓ-diversity problem. In Section 2 we also explain that Anatomy [20], which is an ℓ-diversity-specific method, exhibits high information loss, despite relaxing the privacy requirements (i.e., it publishes the *exact* quasi-identifier). Moreover, while our techniques resemble clustering, our experiments show that existing clustering-based anonymization techniques (e.g., Ref. [21]) are worse in terms of information loss and they are considerably slower.

Summarizing, in this paper we present a framework for solving efficiently the k-anonymity and ℓ-diversity problems, by mapping the multi-dimensional quasi-identifiers to 1-D space. Specifically: *(i)* For k-anonymity, we develop an optimal algorithm for 1-D quasi-identifiers with running time linear to the size of the dataset. *(ii)* For the more complex ℓ-diversity problem, we study theoretically the properties of possible optimal 1-D solutions. Guided by these properties, we propose an efficient heuristic algorithm with linear-time complexity. *(iii)* We generalize our algorithms to multi-dimensional quasi-identifiers, by mapping them to 1-D space. Given a sorted input, the I/O cost is very low, since our algorithms scan the data only once. As a case study, we consider mappings based on the Hilbert [15] space-filling curve and iDistance [23]. *(iv)* The experimental results show that our algorithms clearly outperform the existing state-of-the-art in terms of information loss and running time.

The rest of this paper is organized as follows: Section 2 contains essential definitions and surveys the related work. Section 3 and Section 4 present our solutions for the k-anonymity and ℓ-diversity problem, respectively. In Section 5 we extend our algorithms to the general case of multi-dimensional quasi-identifiers. We present our experimental evaluation in Section 6 and the conclusions in Section 7.

2. BACKGROUND AND RELATED WORK

In this section, we introduce the data model and terminology used in the paper and present the related work.

Definition 1 (Quasi-identifier). *Given a database table $T(A_1, A_2, \ldots, A_n)$, a quasi-identifier attribute set $Q_T = \{A_1, A_2, \ldots, A_d\} \subseteq \{A_1, A_2, \ldots, A_n\}$ is a minimal set of attributes, which can be joined with external information in order to reveal the personal identity of individual records [17].*

A set of tuples which are indistinguishable in the projection of T on Q_T is called *equivalence class.* Two commonly employed techniques to preserve privacy are generalization and suppression [17]. Generalization defines equivalence classes for tuples as multi-dimensional ranges in the Q_T space, and replaces their actual Q_T values with a representative value of the whole range of the equivalent class (e.g., replaces the city with the state). Generalization ranges are usually specified by a generalization hierarchy, or taxonomy tree (e.g., city→state→country). Suppression excludes some Q_T attributes or entire records (known as outliers) from the microdata, altogether.

The privacy-preserving transformation of the microdata is referred to as *recoding*. Two models exist: in *global recoding*, a particular detailed value must be mapped to the same generalized value in all records. *Local recoding*, on the other hand, allows the same detailed value to be mapped to different generalized values in each equivalence class. Local recoding is more flexible and has the potential to achieve lower information loss [10]. The recoding process can also be classified into *single-dimensional*, where the mapping is performed for each attribute individually, and *multi-dimensional*, which maps the Cartesian product of multiple attributes. Multi-dimensional mappings are more accurate; nevertheless initial research efforts focused on single-dimensional ones due to simplicity. In this paper we develop local recoding, multi-dimensional transformations.

All privacy-preserving transformations cause information loss, which must be minimized in order to maintain the ability to extract meaningful information from the published data. Below we discuss suitable information loss metrics.

2.1 Information Loss Metrics

A variety of information loss metrics have been proposed. The *Classification Metric* (CM) [7] is suitable when the purpose of the anonymized data is to train a classifier. Each record is assigned a class label, and information loss is computed based on the adherence of a tuple to the majority class of its group. However, it is not clear how CM can be extended to support general purpose applications. The *Discernibility Metric* (DM) [3], on the other hand, measures the cardinality of the equivalence class. Although classes with few records are desirable, DM does not capture the distribution of records in the Q_T space. More accurate is the *Generalized Loss Metric* [7] and the similar *Normalized Certainty Penalty* (NCP) [21]. The latter factors in both the cardinality of each class and the extent in the Q_T space. For numerical attributes, the NCP of an equivalence class G is defined as:

$$NCP_{A_{Num}}(G) = \frac{max^G_{A_{Num}} - min^G_{A_{Num}}}{max_{A_{Num}} - min_{A_{Num}}}$$

where the numerator and denominator represent the ranges of attribute A_{Num} for the class G and the entire table, respectively. In the case of categorical attributes, where no total order or distance function exists, NCP is defined with respect to the taxonomy tree of the attribute:

$$NCP_{A_{Cat}}(G) = \begin{cases} 0, & card(u) = 1 \\ card(u)/|A_{Cat}|, & otherwise \end{cases}$$

where u is the lowest common ancestor of all A_{Cat} values included in G, $card(u)$ is the number of leaves (i.e., attribute values) in the subtree of u, and $|A_{Cat}|$ is the total number of distinct A_{Cat} values. The NCP of class G over all quasi-identifier attributes is:

$$NCP(G) = \sum_{i=1}^{d} w_i \cdot NCP_{A_i}(G) \qquad (1)$$

where d is the number of attributes in Q_T (i.e., dimensionality). A_i is either a numerical or categorical attribute and has a weight w_i, where $\sum w_i = 1$.

NCP measures information loss for a single equivalence class. Based on it we introduce a new metric, called *Global Certainty Penalty* (GCP), which measures the information loss of the entire anonymized table. Let $\mathcal{P}$ be the set of all equivalence classes in the released anonymized table. GCP is defined as:

$$GCP(\mathcal{P}) = \frac{\sum_{G \in \mathcal{P}} |G| \cdot NCP(G)}{d \cdot N} \qquad (2)$$

where N denotes the number of records in the original table (i.e., microdata) and d is the dimensionality of Q_T. The advantage of this formulation is its ability to measure information loss among tables with varying cardinality and dimensionality. Furthermore, GCP is between 0 and 1, where 0 signifies no information loss (i.e., the original microdata) and 1 corresponds to total information loss (i.e., there is only one equivalence class covering all records in the table).

2.2 k-anonymity

Definition 2 (k-anonymity). *A database table T with a quasi-identifier attribute set Q_T conforms to the k-anonymity property, if and only if each unique tuple in the projection of T on Q_T occurs at least k times [17].*

An *optimal* solution to the k-anonymity problem should minimize information loss. Formally:

Problem 1. *Given a table T, a quasi-identifier set Q_T and a privacy bound expressed as the degree of anonymity k, determine a partitioning $\mathcal{P}$ of T such that each partition $G \in \mathcal{P}$ has at least k records, and $GCP(\mathcal{P})$ is minimized.*

Meyerson and Williams [14] proved that optimal k-anonymity for multi-dimensional quasi-identifiers is NP-hard, under both the generalization and suppression models. For the latter, they proposed an approximate algorithm that minimizes the number of suppressed values; the approximation bound is $O(k \cdot logk)$. Aggarwal et al. [2] improved this bound to $O(k)$. Several approaches limit the search space by considering only global recoding. Ref. [3] proposes an optimal algorithm for single-dimensional recoding with respect to the CM and DM metrics. *Incognito* [9] introduces a dynamic programming approach, which finds an optimal solution for any metric by considering all possible generalizations (only for global, single-dimensional recoding).

To address the inflexibility of single-dimensional recoding, *Mondrian* [10] employs multi-dimensional global recoding, which achieves finer granularity. Similar to *kd*-trees [4], Mondrian partitions the space recursively across the dimension with the widest normalized range of values. Mondrian can also support a *limited* version of local recoding: if many points fall on the boundary of two groups, they may be divided between the two groups. Because Mondrian uses space partitioning, the data points within a group are not necessarily close to each other in the Q_T space (e.g., points 22 and 55 in Figure 1(b)), causing high information loss.

Another family of multi-dimensional local recoding methods is based on clustering. In Ref. [1] k-anonymity is treated as a special clustering problem, called r-cellular clustering. A constant factor approximation of the optimal solution is proposed, but the bound only holds for the Euclidean distance metric. Furthermore, the computation and I/O cost may be high in practice. Ref. [21] proposes agglomerative and divisive recursive clustering algorithms, which attempt to minimize the NCP metric. The latter (called *TopDown* in the following) is the best of the two. TopDown performs a two-step clustering: first, all records are in one cluster, which is recursively divided as long as there are at least $2k$ records in each cluster. In the second step, the clusters with less than k members are either grouped together, or they borrow records from clusters with more than k records. The complexity of TopDown is $O(N^2)$. In our experiments, we show that TopDown is inefficient in terms of information loss and computational cost.

2.3 ℓ-diversity

A database table T with a quasi-identifier attribute set Q_T and a *Sensitive Attribute* SA, conforms to the ℓ-diversity property, if and only if each equivalence class in T with respect to Q_T has at least ℓ well-represented values for the attribute. Ref. [13] proposed two interpretations of "well-represented values": *entropy* ℓ-diversity and *recursive (c,ℓ)-diversity.* The former yields tighter privacy constraints, but is too restrictive for practical purposes. The latter proposes a more relaxed condition: An equivalence class G is ℓ-diverse if $f_1 < c(f_\ell + f_{\ell+1} + .. + f_m)$, where c is a constant, f_i is the number of occurrences of the i^{th} most frequent value of SA in G, and m is the number of distinct values in SA. In order for an ℓ-diverse partitioning to exist, the original table T must itself satisfy the above condition, referred to as the *eligibility condition* (EG).

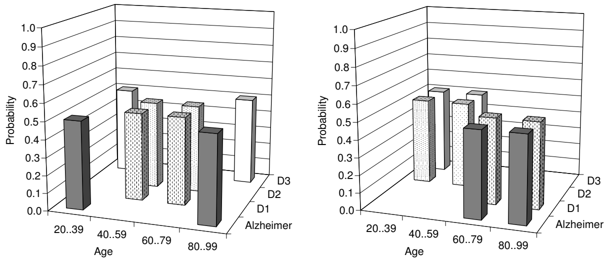
In practice, the privacy threat to a certain database record is expressed as the probability of associating an anonymized record with a certain value $s \in SA$; we denote this breach probability by P_{br}. Note that, given an equivalence class G, $P_{br} = \#occurrences(s)/|G|$. Since P_{br} is directly relevant to the privacy of records, it is desirable to have an ℓ-diversity formulation that can be linked to P_{br}. We therefore adopt the following ℓ-diversity formulation from Ref. [20]:

DEFINITION 3 (ℓ-DIVERSITY). *An equivalence class G has the ℓ-diversity property, if the probability of associating a record in G with any particular sensitive attribute value is at most 1/ℓ.*

Under this definition, the eligibility condition requires that at most $|T|/\ell$ tuples in the original table T have the same SA value. An *optimal* solution to the ℓ-diversity problem should minimize information loss. Formally:

PROBLEM 2. *Given a table T, a quasi-identifier Q_T and a privacy bound expressed as the degree of diversity ℓ, determine a partitioning $\mathcal{P}$ of T such that each partition $G \in \mathcal{P}$ satisfies the ℓ-diversity property and $GCP(\mathcal{P})$ is minimized.*

Ref. [13] implements ℓ-diversity on top of Incognito and suggests that any k-anonymity technique can be adapted for ℓ-diversity. However, as we demonstrate in the example of Figure 1, k-anonymity techniques may result to unacceptable information loss, due to the requirement of diverse SA values.



(a) Anatomy

(b) Our Approach

Figure 2: ℓ-diversity example. In the microdata only the 80 years old group can suffer from Alzheimer's.

Anatomy [20] is an ℓ-diversity-specific method. It hashes records into buckets according to the SA value, and builds partitions by randomly selecting ℓ records from distinct buckets; the complexity is $O(|T|)$. Anatomy has two drawbacks: *(i)* It releases the *exact* quasi-identifiers of records. While this does not violate the ℓ-diversity property, it confirms that a particular individual is included in the data. Consider for instance a dataset containing quasi-identifiers of convicted persons and their crime. Although Anatomy hides the exact crime, an attacker can still conclude that a specific person has been convicted. Therefore, Anatomy fails to provide the very basic anonymity safeguard for which the k-anonymity model was proposed in the first place. *(ii)* Anatomy does not consider the partition's extent in the Q_T space, hence information loss may be high. Consider a medical dataset in which all persons who suffer from Alzheimer's are more than 80 years old; assume $\ell = 2$. Anatomy may choose to group together records from the [20..39] and [80..99] age intervals (Figure 2(a)), suggesting that Alzheimer's is equally probable for young and old persons. In contrast, our approach generates the [60..99] group (Figure 2(b)), and correctly implies that Alzheimer's is only possible for elderly patients.

Like Anatomy, Ref. [22] publishes the exact Q_T. It focuses on SAs with numerical values and deals with situations where these values are similar; its drawbacks are analogous to Anatomy's. Another recent work [12] proposes a new privacy paradigm called *t-closeness*, which dictates that the table-wise distribution of SA values should be reproduced within each anonymized group. No specific technique is proposed; instead, it is suggested to modify existing k-anonymity techniques. However, this is expected to face the same drawbacks as the application of k-anonymity techniques to ℓ-diversity. Yet another model is described in Ref. [18], where each record in the table has an individual privacy constraint. However, in order to enforce privacy, SA values must also be generalized. Ref. [8] proposes a method for publishing anonymized marginals, in addition to microdata. Marginals are summaries of the original table that may improve accuracy. Anonymizing the marginals is orthogonal to anonymizing the microdata.

3. OPTIMAL 1-D K-ANONYMITY

In this section we present an optimal solution to the k-anonymity problem for one-dimensional quasi-identifiers. Al-

```
**Optimal 1-D $k$-anonymity**
Input: set $\mathcal{R}$ in ascending order of 1-D $Q_T$
1.  for $i := k$ to $2k-1$
2.    $Opt(i) = Opt_I([1,i])$
3.    $prev(i) = NIL$ /* used to reconstruct solution*/
4.  for $i := 2k$ to $N$
5.    for $j := \max\{k, i-2k+1\}$ to $i-k$
6.      $Opt(i) = \min_j \{Opt(j) + Opt_I([j+1,i])\}$
7.      $prev(i) = j$ value that minimizes $Opt(i)$
8.  $i = N$ /* output $k$-anonymized groups */
9.  while $(prev(i) \neq NIL)$
10.   output group with boundaries $[prev(i)+1, i]$
11.   $i = prev(i)$
12. output group $[1,i]$
```

Figure 3: Pseudocode for optimal 1-D k-anonymity

though the problem is NP-hard in the general case [14], we show that the complexity is linear to the size of the input table for 1-D quasi-identifiers[2].

Let $\mathcal{R} = \{r_i\}_{1 \leq i \leq N}$ be the set of records in table T, where $N = |T|$. $\mathcal{R}$ is a *totally ordered* set according to the 1-D quasi-identifier Q_T. Our goal is to compute a partitioning of $\mathcal{R}$ that minimizes GCP and satisfies the k-anonymity property.

An algorithm that computes the 1-D optimal k-anonymity partitioning of $\mathcal{R}$ needs only to consider groups with records that are consecutive in the Q_T space. This results immediately from the fact that if two groups with at least k records each overlap, we can swap records between them such that, the number of records in each group remains the same and the overlap is eliminated, without increasing GCP.

The weighted NCP metric (which is a component of GCP) is *superadditive*: Given an equivalence class G and two subsets G_1 and G_2 such that $G = G_1 \cup G_2$ and $G_1 \cap G_2 = \emptyset$, then $|G| \cdot NCP(G) \geq |G_1| \cdot NCP(G_1) + |G_2| \cdot NCP(G_2)$. This is due to the fact that the total set of records remains the same and the extent of G_1 plus G_2 in the Q_T space cannot exceed that of G (since there are no overlaps). It is straightforward to show that GCP is also superadditive.

LEMMA 1. *Let $\mathcal{P}$ be the optimal k-anonymity partitioning of a set $\mathcal{R}$ according to GCP. Then $\mathcal{P}$ does not contain groups of more than $2k-1$ records.*

PROOF. Assume that a group G in $\mathcal{P}$ contains more than $2k-1$ records. We split G into two groups G_1 and G_2 of at least k records each, such that $G = G_1 \cup G_2$, $G_1 \cap G_2 = \emptyset$. Since GCP is superadditive, $GCP(\mathcal{P}) \geq GCP((\mathcal{P} \setminus G) \cup G_1 \cup G_2)$; hence information loss cannot increase. Therefore the optimal partitioning does not need to contain groups of cardinality larger than $2k-1$. □

The 1-D k-anonymity problem under the GCP metric can be solved with dynamic programming as follows: Let $Opt(i)$ be the information loss of the optimal partitioning achieved for the prefix subset of the first i records of $\mathcal{R}$; and $Opt_I([b,e]) = (e-b+1) \cdot NCP(\{r_b, \ldots, r_e\})$ be the information loss of the group containing all records in the interval $\{r_b, \ldots, r_e\}$. Then:

$$Opt(i) = \min_{i-2k<j\leq i-k} (Opt(j) + Opt_I([j+1,i]))$$

This recursive scheme selects the best out of all suffixes of $\mathcal{R}$ to create the next group. Since every group should contain between k and $2k-1$ records, it follows that the end boundary record of the previously created group must be in the interval $[i-2k+1, i-k]$. The optimal solution for all j-prefixes of $\mathcal{R}$, $k \leq j \leq 2k-1$, is computed directly. Then, the computation proceeds with increasing i, $2k \leq i \leq N$. The pseudo-code for the algorithm is given in Figure 3. The algorithm generates an optimal partitioning $\mathcal{P}$. Note that $GCP(\mathcal{P}) = Opt(N)/N$.

Complexity Analysis. The algorithm ranges through $O(k)$ values of j for $O(N)$ values of i. If $Opt_I([j+1,i])$ can be computed in $O(\omega)$, then the time complexity is $O(kN\omega)$. The dynamic programming array has N entries; however, we only need to access a constant fraction $O(k)$ of the array at any one time. After the computation ends, we must scan the entire array (lines 9-11) at most one more time to output the solution, unless the complete array fits in the main memory.

4. ONE-DIMENSIONAL ℓ-DIVERSITY

In this section we study the ℓ-diversity problem for one-dimensional quasi-identifiers. In contrast to k-anonymity, optimal solutions for ℓ-diversity cannot be computed efficiently even for 1-D quasi-identifiers. The inefficiency arises from the fact that the resulting partitioning may contain overlapping groups; therefore, numerous possible combinations must be examined. In this section, we study theoretically the properties of an optimal solution. Guided by these properties, we develop an efficient linear-time (to the size of the input) heuristic algorithm.

To simplify our theoretical investigation, in Section 4.1 we use a simplified information loss metric[3]:

$$\mathcal{IL}(\mathcal{P}) = \max_{G \in \mathcal{P}} (max^G_{Q_T} - min^G_{Q_T}) \qquad (3)$$

which represents the maximum extent (in the 1-D quasi-identifier space) of any group G in partitioning $\mathcal{P}$. $\mathcal{IL}$ is superadditive: $\mathcal{IL}(G_1 \cup G_2) \geq \max(\mathcal{IL}(G_1), \mathcal{IL}(G_2))$.

4.1 Properties of the Optimal Solution

Let $\mathcal{R} = \{r_i\}_{1 \leq i \leq N}$ be the set of records in the original table, and S the projection of $\mathcal{R}$ on the sensitive attribute (SA). Denote by $r_i.Q$ the 1-D Q_T value of r_i and by $r_i.S$ the SA value of record r_i. Let $m = |S|$, i.e., there are m distinct values of SA. For a pair of records r_i, r_j we denote $|r_i - r_j| = |r_i.Q - r_j.Q|$.

LEMMA 2. *Let $\mathcal{P}$ be an optimal ℓ-diversity partitioning of $\mathcal{R}$ according to the information loss metric $\mathcal{IL}$. Then $\mathcal{P}$ does not need to contain groups of more than $2\ell-1$ records.*

PROOF. Assume there is a group G in the optimal solution such that $|G| \geq 2\ell$. We can express the cardinality of G as $|G| = c \cdot \ell + r, c \geq 2, 0 \leq r < \ell$. Since G is ℓ-diverse, according to Definition 3 every SA value in G can occur at most c times. There are at most ℓ values in G with c occurrences. We remove from G the ℓ records with the most frequent SA values in G, and create group G'. By construction, G' is ℓ-diverse. Let $G'' = G \setminus G'$. Any sensitive attribute value in G'' can occur at most $c-1$ times and $|G''| = (c-1)\ell + r$. Hence, G'' is ℓ-diverse. Furthermore, since $\mathcal{IL}$ is superadditive, $\mathcal{IL}(\mathcal{P}) \geq \mathcal{IL}((\mathcal{P} \setminus \{G\}) \cup \{G'\} \cup \{G''\})$. Splitting

[2] A similar solution has been independently developed in [19] and applied for multi-dimensional quasi-identifiers, after mapping them to the 1D space using space-filling curves

[3] We discuss the GCP metric in Section 5.

G'' recursively, we obtain a partitioning with equal or improved utility compared to $\mathcal{P}$, and cardinality of each group between ℓ and $2\ell - 1$. $\square$

COROLLARY 1. **Value Singularity Property** *In an optimal ℓ-diverse partitioning $\mathcal{P}$, every group $G \in \mathcal{P}$ contains at most one occurrence for any SA value $s_j \in S$.*

PROOF. Assume an optimal solution $\mathcal{P}$ and $G \in \mathcal{P}$ such that s_j appears twice in G. Since $|G| \leq 2\ell - 1$, it results that G is not ℓ-diverse, i.e., a contradiction. $\square$

Since there are only m distinct SA values, we conclude that $|G| \leq min(2\ell - 1, m)$.

$\mathcal{R}$ is a *totally ordered* set according to Q_T, and each record in $\mathcal{R}$ belongs to exactly one ℓ-diverse group. Hence, the first (i.e., b_i) and last (i.e., e_i) record in group G_i are defined according to the total order in $\mathcal{R}$; we call b_i the *begin* and e_i the *end* record of G_i. Hence there exists a total order of both begin and end records of the set of groups in the optimal solution. We refer to the b_i and e_i records as *border* elements. Note that, unlike the case of k-anonymity, a group need not contain only consecutive records in the ordering.

Let *domain* $\mathcal{D}_q = \{r_i \in \mathcal{R} | r_i.S = s_q\}, 1 \leq q \leq m$. Figure 4 depicts the domains $\mathcal{D}_q$ for a 3-diverse partitioning of $\mathcal{R}$, $m = 4$. Note that, the total order in the quasi-identifier space induces a total order for each of the domains $\mathcal{D}_q$.

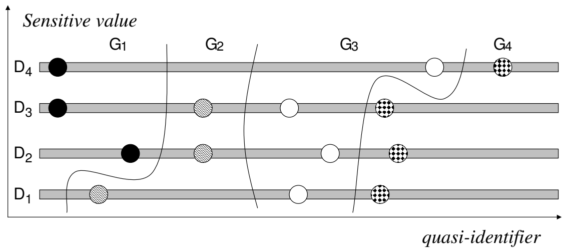


Figure 4: Sensitive Value Domains

The following lemma shows that the order of appearance of groups in each value domain $\mathcal{D}_q$ is the same.

LEMMA 3. **Group Order Property** *There exists an optimal ℓ-diverse partitioning $\mathcal{P}$ of $\mathcal{R}$, producing $|\mathcal{P}|$ groups $G_1, G_2, \ldots G_{|\mathcal{P}|}$, such that the order of sets $\{G_1^q, G_2^q, \ldots, G_{|\mathcal{P}|}^q\}$, defined for the groups in $\mathcal{P}$ as they appear along each domain $\mathcal{D}_q$, $G_i = \cup_q G_i^q$, $1 \leq i \leq |\mathcal{P}|$, is consistent across all domains $\mathcal{D}_q$, $1 \leq q \leq m$ (except for the fact that some groups may not be represented in each domain).*

PROOF. Assume an optimal solution in which there exist records $r_i \in G_i^q$ and $r_j \in G_j^q$ such that $r_i.Q < r_j.Q$, and records $t_i \in G_i^p$ and $t_j \in G_j^p$ such that $t_j.Q < t_i.Q$. Then, for all possible relative orderings in the one-dimensional Q_T, $|r_i - t_j| + |r_j - t_i| \leq |r_i - t_i| + |r_j - t_j|$. Let $G_i' = G_i \diagdown \{t_i\} \cup \{t_j\}$ and $G_j' = G_j \diagdown \{t_j\} \cup \{t_i\}$. Then it results that $\mathcal{IL}(G_i') + \mathcal{IL}(G_j') \leq \mathcal{IL}(G_i) + \mathcal{IL}(G_j)$, i.e. $\mathcal{IL}(\mathcal{P})$ cannot be increased by exchanging t_j and t_i. Since t_i and t_j have the same SA value, the ℓ-diversity of the partitioning is not affected by the exchange. The same reasoning can be applied for all remaining pairs of records that violate a given order. It follows that the order of the partitions in the newly constructed optimal partitioning is consistent across all domains $\mathcal{D}_q$, $1 \leq q \leq m$. $\square$

We write $G_i \prec G_j$ to denote that G_i precedes G_j in the partial order defined over optimal partitioning $\mathcal{P}$. As a consequence of Lemma 3, in order to find an optimal solution, we can build groups by assigning records from each domain *in order*. This prunes significantly the search space of the solution. Figure 5 shows an example where the Group Order Property is violated. Let $G_1 = \{r_1, r_3, r_5\}$ and $G_2 = \{r_2, r_4, r_6\}$. G_1 precedes G_2 in the D_3 domain, while the opposite occurs for D_2. However, the optimal solution is $G_1' = \{r_1, r_2, r_4\}$, $G_2' = \{r_3, r_5, r_6\}$ and $G_1' \prec G_2'$.

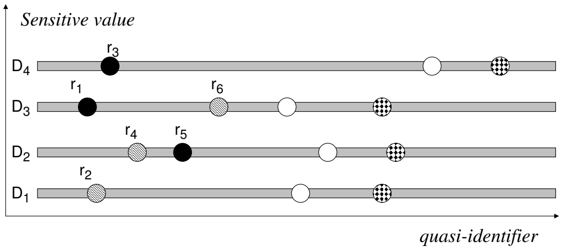


Figure 5: Group Order Violation

The group ordering extends to the begin and end records of groups, as proved by the following lemma:

LEMMA 4. **Border Order Property** *There exists an optimal ℓ-diverse partitioning $\mathcal{P}$ of $\mathcal{R}$, producing $|\mathcal{P}|$ groups $G_1, G_2, \ldots G_{|\mathcal{P}|}$ with begin records $b_1, b_2, \ldots, b_{|\mathcal{P}|}$ and end records $e_1, e_2, \ldots, e_{|\mathcal{P}|}$, such that the begin and end sets obey the same order as the groups $\{G_1, G_2, \ldots, G_{|\mathcal{P}|}\}$ they belong to, i.e. if $G_i \prec G_j$, then $b_i.Q < b_j.Q$ and $e_i.Q < e_j.Q$.*

PROOF. The proof is similar to that of Lemma 3. $\square$

Lemma 4 further reduces the search space by limiting the choices of records for the currently built group based on the begin and end records of the previously built group. Figure 6 shows an example where the Border Order Property is violated (although the Group Order Property is satisfied). Let $G_1 = \{r_1, r_2, r_6\}$ and $G_2 = \{r_3, r_4, r_5\}$. b_1 (i.e., r_1) precedes b_2 (i.e., r_3), but e_1 (i.e., r_6) succeeds e_2 (i.e., r_5). The solution is not optimal; in the optimal case, $G_1' = \{r_1, r_2, r_3\}$, $G_2' = \{r_4, r_5, r_6\}$, $b_1' < b_2'$ and $e_1' < e_2'$.

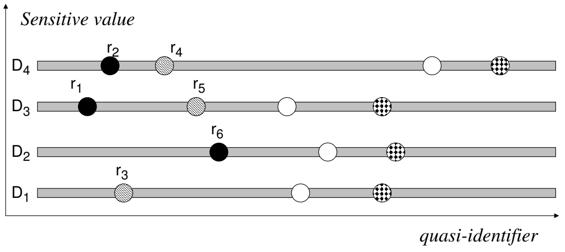


Figure 6: Border Order Violation

LEMMA 5. **Cover Property** *There exists an optimal ℓ-diverse partitioning $\mathcal{P}$ of $\mathcal{R}$ with the following property: $\forall G_i$, $G_j \in \mathcal{P}$ such that $G_i \prec G_j$, and $\nexists G_k : G_i \prec G_k \prec G_j$, if there exists a pair of records $r \in G_i, t \in G_j$, such that $r.Q > t.Q$, then there is either a record $r' \in G_j$ of the same sensitive value as r (where, according to Lemma 3, $r'.Q > r.Q$) or a record $t' \in G_i$ of the same sensitive value as t (where, according to Lemma 3, $t'.Q < t.Q$), or both.*

PROOF. Assume there are records $r \in G_i$, $t \in G_j$ such that $r.Q > t.Q$, and there is neither $r' \in G_j$ with same SA value as r, nor $t' \in G_i$ with same SA value as t. Then we can swap r and t between G_i and G_j, without compromising ℓ-diversity. Furthermore, since $b_i.Q \leq b_j.Q \leq t.Q \leq r.Q \leq e_i.Q \leq e_j.Q$, it follows that the swap does not increase the information loss. Hence, we have obtained an optimal solution, where $\nexists(r \in G_i, t \in G_j)$ such that $r.Q > t.Q$. □

The intuition behind the Cover Property is that if record r can be added to any of two groups G_1 and G_2, then it should be added to the group that is closer to r in the Q_T space. Figure 7 shows an example where the Cover Property is violated: consider partially completed groups $G_1 = \{r_1, r_3\}$ and $G_2 = \{r_5, r_6\}$. If r_2 is assigned to G_2 and r_4 to G_1, the Cover Property does not hold; in an optimal solution, r_4 must belong to G_2 and r_2 to G_1.

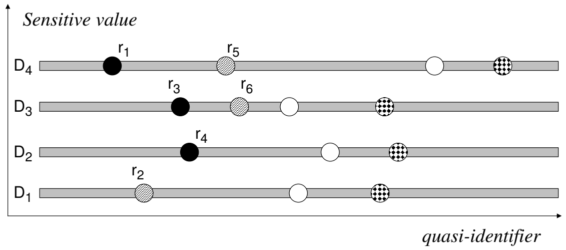


Figure 7: Cover Property Violation

The *end boundary* $\mathbf{E}_i$ of a group G_i is a vector $\{e_1, \ldots e_m\}$, where each e_q corresponds to one SA value. Intuitively, $\mathbf{E}_i$ marks the position of the last record of G_i in each domain D_q of a sensitive value s_q. If G_i does not contain a record with sensitive value $s_{q'}$, then $e_{q'}$ is equal to the corresponding $e_{q'}$ in the previous group G_j ($G_j \prec G_i$). The delimitation lines in Figure 4 give an intuitive interpretation of boundaries. For the second group (G_2), for instance, $\mathbf{E}_2 = \{1, 2, 2, 1\}$. As an immediate result of Lemma 4, if two groups are ordered as $G_j \prec G_i$, then the same order is enforced for their corresponding end boundaries. In other words, even if groups overlap in the Q_T space, their boundaries defined over the D_q domains *do not overlap.*

By considering all possible boundaries, we can show that an optimal solution can be found with $O(2^m N^{m+2})$ worst-case complexity. Obviously, such an approach is prohibitive in practice, even for small inputs, as the value of m can be quite large.

4.2 Efficient 1-D ℓ-diversity Heuristic

We present a heuristic 1-D ℓ-diversity algorithm which is inspired by the theoretical analysis of the previous section. Given a sorted input, our algorithm exhibits linear time and I/O cost in the size of the input table. The algorithm guarantees that if the original table satisfies the eligibility condition (EG, see Section 2.3) for a given ℓ value, then a solution will be found, although it may not be an optimal one.

First, the records are sorted according to their Q_T value and are assigned to m domains $D_{1 \leq q \leq m}$, based on their sensitive attribute value. Subsequently, following the results from Corollary 1 and Lemma 3 and 4, the group formation phase attempts to form groups having between ℓ and m records with distinct SA values. Let $\mathbf{E} = \{e_1, e_2, \ldots e_m\}$ be the end boundary of the previously formed group. We denote by *frontier* of the search the set $\{r_q \in D_q | 1 \leq q \leq m\}$, such that each r_q is the successor of e_q in its respective domain. Initially, the frontier consists of the first record in each domain D_q.

```
Heuristic 1-D $\ell$-diversity
Input: set $\mathcal{R} = \{r_i\}_{1 \leq i \leq N}$ in ascending order of 1-D $Q_T$
split sorted records in $m$ buckets based on SA value
$H[i]$ = #records in bucket $i$
$remaining = N$
frontier $\mathcal{F}$ = {set of first record in each bucket}
while ($remaining > 0$)
  $count = \ell$
  do /*greedy step*/
    $G$ = {set of $count$ records of $\mathcal{F}$ with lowest $Q_T$}
    $count++$
  until ($EG$ holds or $count > m$)
  if ($EG$ does not hold) /*fall-back step*/
    $count = \ell$
    do
      $G$ = {set of $count$ records in $\mathcal{F}$ with max $H$ value}
      $count++$
    until ($EG$ holds)
  close $G$, update $H$ and advance $\mathcal{F}$
  $remaining = remaining - count + 1$
output $\ell$-diverse groups
```

Figure 8: Heuristic 1-D ℓ-diversity

The heuristic consists of two steps: the *greedy step* and the *fall-back step*. In the greedy step, it assigns to the current group G the ℓ records on the frontier with the lowest Q_T values, and checks if eligibility condition EG holds for the remaining records. If EG is satisfied, then G is closed, the frontier is advanced beyond the records in G, and the algorithm starts building the next group. Otherwise, out of the remaining unassigned records on the frontier, the record with the lowest Q_T is added to G, and EG is re-evaluated. The process continues until EG is satisfied, or all m records on the frontier are in G.

If EG is still not satisfied, the records in G are rolled back, and the following fall-back strategy step is executed: ℓ of the records on the frontier with SA values which are the most frequent among the *unassigned* records are added to G (in case of ties, the record with the lowest Q_T is chosen). Then, EG is evaluated, and if it does not hold, the record with the $(\ell+1)^{th}$ most frequent value is added, and so forth, up to $m-1$ (the case where all m records on the frontier are chosen has been considered in the greedy step). It is guaranteed that, by picking the most frequent records the EG is eventually satisfied [20]; therefore, a solution can be found. Note that, the fact that the fall-back step is executed for the current group does not imply that it will be necessary for the next one. The fall-back step may be necessary for Q_T regions with significant variance in the density of records with distinct SA values.

Figure 8 shows the pseudocode for the heuristic algorithm. To evaluate EG, we maintain the counter *remaining* with the number of unassigned records, and a histogram H with the distribution of remaining records to SA values. Upon each record assignment, *remaining* and H are updated. The histogram can easily fit into memory (it contains m elements); the updates and EG evaluation cost $O(m)$.

The presented heuristic will finalize the current group G if it is able to find $count \leq m$ records such that EG holds. However, in some cases, this approach may generate groups with large extent. Consider the example in Figure 9, where ℓ=3. After picking the first three records, the algorithm closes G at boundary 1, and r_4 is grouped with r_{5-7}. How-

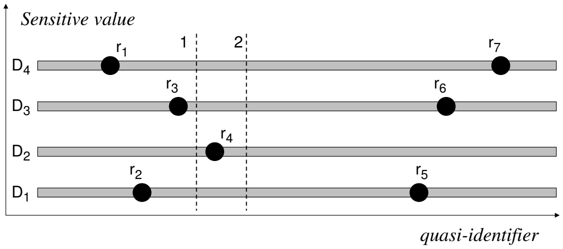


Figure 9: Heuristic Optimization, ℓ=3

ever, if it were grouped with r_{1-3} (boundary 2), the extent of the partitioning (hence, the resulting information loss) would be considerably smaller.

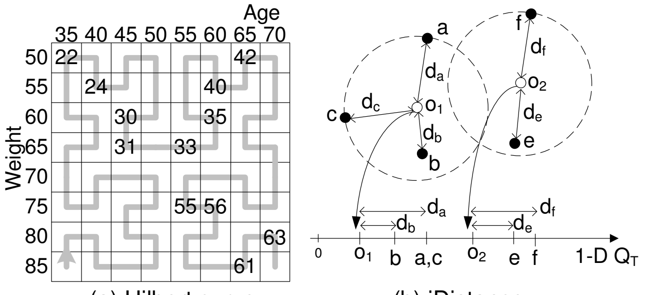
As an enhancement to our heuristic, we propose the following optimization: after G is formed (e.g., $\{r_1, r_2, r_3\}$ in Figure 9), we inspect records r_A and r_B on the frontier with the 1^{st} and respectively ℓ^{th} lowest Q_T value (i.e., $r_A \equiv r_4$, $r_B \equiv r_6$ in the example). The extent of the group that contains $r_A, .., r_B$ is a lower bound for the extent of the group that will contain r_A. If the distance from r_A to the leftmost record in G (e.g., $|r_4 - r_1|$) is smaller than the distance from r_A to r_B (e.g., $|r_4 - r_6|$), and there is not already a record with $r_A.S$ in G (e.g., no record from D_2 in G), we add r_A to G, subject to EG being satisfied. In the running example, the two obtained groups are $\{r_1, r_2, r_3, r_4\}$ and $\{r_5, r_6, r_7\}$. This optimization aims at reducing the information loss of ℓ-diverse groups, and has complexity $O(1)$.

5. GENERAL MULTI-DIMENSIONAL CASE

In this section we extend our 1-D k-anonymity and ℓ-diversity algorithms to multi-dimensional quasi-identifiers. Let Q_T be a quasi-identifier with d attributes (i.e., d dimensions). We map the d-dimensional Q_T to one dimension and execute our 1-D algorithms on the transformed data. Recall that both optimal k-anonymity and ℓ-diversity are NP-hard [14, 13] in the multi-dimensional case. The solutions we obtain through mapping are not optimal; however, due to the good locality properties of the space mapping techniques, information loss is low, as we demonstrate experimentally in Section 6. In the following, we measure the information loss of each k-anonymous or ℓ-diverse group using NCP, and the information loss over the entire partitioning using GCP (see Section 2).

We employ two well-known space-mapping techniques: the *Hilbert space-filling curve* [15] and *iDistance* [23]. The Hilbert curve is a continuous fractal which maps each region of the space to an integer. With high probability, if two points are close in the multi-dimensional space, they will also be close in the Hilbert transformation [15]. Figure 10(a), for instance, shows the transformation from 2-D to 1-D for the 8 × 8 grid of the example in Section 1; the granularity of the regions can be arbitrarily small. The data set is totally ordered with respect to the 1-D Hilbert value.

iDistance is optimized for nearest-neighbor queries. In iDistance, a random sample of the data is first clustered around a fixed number of center points. The cluster centers are ordered according to any method (e.g., Hilbert ordering). Each data point is then assigned to its closest cluster center according to Euclidean distance. The 1-D value of a point p is the sum of the 1-D value of its cluster center C, plus the distance from p to C (see Figure 10(b)).



(a) Hilbert curve (b) iDistance

Figure 10: Multi-dimensional to 1-D mappings

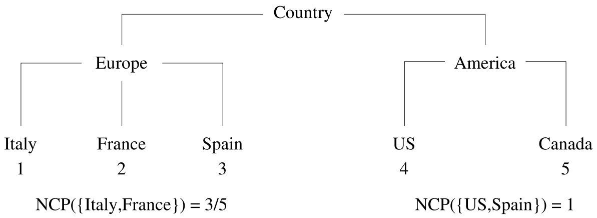


Figure 11: Categorical Attribute Mapping

Regardless of the technique, in order to perform the data mapping, each attribute value must be assigned to a number. For numerical attributes, we can use the attribute value directly; furthermore, the semantic distance between two numeric attribute values can be measured as the difference between the two values. For categorical attributes and their associated taxonomy tree, we adopt the labeling approach of Ref. [3, 9], where each attribute value is assigned to a distinct integer according to the in-order traversal of the taxonomy tree. If an equivalence class spans across different subtrees, it is penalized according to NCP. Figure 11 shows an example, where $NCP(Italy, France) = 3/5$ because their common ancestor is Europe (which has 3 leaves) and there are 5 leaves in the entire Country domain. Also, $NCP(US, Spain) = 1$ (i.e., maximum information loss), because their common ancestor is the entire Country domain. Note that, the mapping is performed only with respect to Q_T; the sensitive attribute is not included in the mapping.

The overhead of the Hilbert mapping is $O(d)$ per record, hence the method is efficient. For iDistance, the mapping involves the additional overhead of finding the c cluster centers for a random sample of the data. In our implementation, we use 10% of the input table to determine the centers, and we set $c = 512$. After selecting the centers, the overhead of mapping is $O(c)$ per record. Our 1-D k-anonymity and ℓ-diversity algorithms require the input to be sorted on Q_T; the cost is $O(N \log N)$. Assuming sorted input, our methods need to scan the data only once; therefore the I/O cost is linear. Below, we discuss some further issues about the extension of our 1-D algorithms to d dimensions.

5.1 k-anonymity-Specific Issues

The k-anonymity dynamic programming algorithm builds two tables: *(i)* the main table with N entries, which stores at entry i the cost of the optimal solution for the first i records, and *(ii)* the auxiliary table that stores the base-case cost (i.e., NCP) for each sequence of consecutive k to $2k - 1$ records. Since the tabulation proceeds from left to right,

Attribute	Cardinality	Type
Age	79	Numerical
Gender	2	Hierarchical (2)
Education Level	17	Numerical
Marital Status	6	Hierarchical (3)
Race	9	Hierarchical (2)
Work Class	10	Hierarchical (4)
Country	83	Hierarchical (3)
Occupation	50	Sensitive Value
Salary Class	50	Sensitive Value

Table 1: CENSUS Dataset Characteristics

at each step we need to look back at most $2k-1$ entries; therefore, we do not need more than a constant fraction of the tables in main memory. If the tables do not fit in main memory, we need to store and then read them from the disk once; the I/O cost is $O(N)$.

The time required to compute the NCP for a sequence of records is linear to the sequence length. Since the sequences are in the form $[r_{i-2k+2}, r_i] \dots [r_{i-k+1}, r_i]$, we optimize this process as follows: for each sequence $[r_a, r_b]$, we use the already computed cost for the sequence $[r_a, r_{b-1}]$, and check if r_b increases the cost. The check needs constant time, if we maintain the *minimum bounding rectangle* (MBR) of each sequence. This reduces the computational cost for the auxiliary table from $O(k^2N)$ to $O(kN)$. Still, updating the MBR and recomputing the NCP costs $O(d)$. To improve execution time, we also implement more time-efficient versions of our algorithms, HilbFast and iDistFast, which calculate the cost of each sequence by its extent in the 1-D space. This variation relies on the assumption that records in close proximity in the multi-dimensional space are also likely to be close in the 1-D space. Hence, there is no need to maintain an auxiliary table at all.

5.2 ℓ-diversity-Specific Issues

During the preprocessing step, our ℓ-diversity algorithm partitions the input into m buckets, one for each value of the sensitive attribute. Combined with the sorting of mapped 1-D data, the preprocessing step costs $O(NlogN)$. Since tabulation is not needed, the space requirement of the algorithm is $O(m)$ (i.e., constant in practice), as we only need to access the frontier of the search at each step, and look back at most one group; this can easily fit in memory. The NCP computation for each ℓ-diverse group formation is $O(m)$.

6. EXPERIMENTAL EVALUATION

In this section, we evaluate our techniques against the existing state-of-the-art. All algorithms are implemented in C++ and the experiments were run on an Intel Xeon 2.8GHz machine with 2.5GB of RAM and Linux OS.

Our workload consists mainly of the CENSUS[4] dataset, containing information of 500,000 persons. The schema is summarized in Table 1: there are nine attributes; the first seven represent the quasi-identifier Q_T, whereas the last two (i.e., *Occupation* and *Salary*) are the sensitive attributes (SA) (for brevity, we only include in our evaluation the *Occupation* attribute). Two of the attributes are numerical and the rest categorical; the number of levels in the taxonomy trees is shown in parentheses. We generate input tables with 50,000 to 400,000 records, by randomly selecting tuples from the entire dataset. We also consider the ADULT dataset, used in previous work [7, 9, 10]. The dataset consists of 30,162 records with eight Q_T attributes, out of which two are numerical and the rest categorical. Due to the small size of ADULT, we use the larger (i.e., more realistic) CENSUS dataset for most of our experiments.

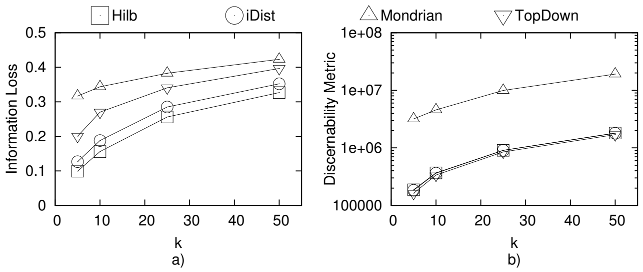


Figure 12: Adult Dataset, variable k

We use the GCP metric (Section 2) to measure the information loss. Recall that the values of GCP are in the range $[0, 1]$, and 0 is the best score (i.e., no information loss).

6.1 k-anonymity Evaluation

In the following experiments, we compare our 1-D optimal k-anonymity algorithm against the existing state-of-the-art techniques: The multi-dimensional *(Mondrian) k*-anonymity [10], and the *TopDown* clustering-based technique [21] (see Section 2 for details). For our optimal 1-D algorithm, we consider both the Hilbert (with 12 bits per dimension) and iDistance (with 512 reference points) mappings. For each of the two mappings, we consider two versions: *(i)* In the base version (i.e., *Hilb* and *iDist*), partitioning is guided by accurate cost estimation at the original multi-dimensional space. As discussed in Section 5, the amortized complexity for calculating the cost is $O(d)$, where d is the dimensionality of Q_T. *(ii)* In the faster variants *HilbFast* and *iDistFast* (see Section 5), the algorithm estimates the cost at the 1-D space in $O(1)$ time. Since this is only an estimation of the real cost, the resulting information loss is expected to be higher.

First, we consider the ADULT dataset and vary k between 5 and 50. We show the information loss in Figure 12(a). Both Hilb and iDist outperform the existing methods. In Figure 12(b) we repeat the experiment using the DM metric, which was also used in the original Mondrian paper. Recall from Section 2 that DM is not particularly accurate to characterize information loss; we include it in our evaluation for illustration purposes only. Even with the DM metric, Mondrian is one order of magnitude worse. Observe that Hilb, iDist and TopDown restrict the partition size between k and $2k-1$. Since DM considers only the partition size, these methods behave similarly, although they are considerably different in terms of information loss. This is another indication that DM is not an appropriate metric.

In the next experiment we use the CENSUS dataset and set the input size $N = 200,000$ records; we vary k from 10 to 100. Figure 13 presents the results. Both Hilb and iDist achieved lower information loss, compared to TopDown and Mondrian, in all cases. In terms of execution time, Mondrian was faster. However, given the superior quality of the results, we believe that the running time of Hilb would be acceptable in practice (it was 60sec in the worst case). iDist is a little slower than Hilb, due to the initial phase of selecting the reference points. Both Hilb and iDist execution times include the data mapping and sorting phase. We also

[4] http://www.ipums.org/

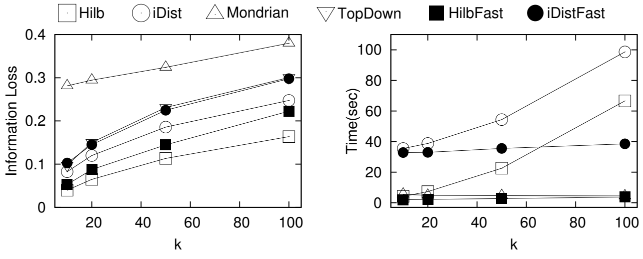


Figure 13: Census Dataset, variable k

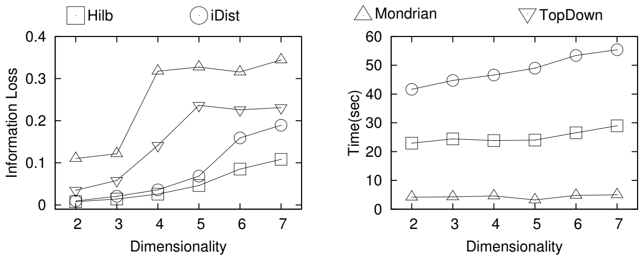


Figure 15: Census Dataset, variable dimensionality

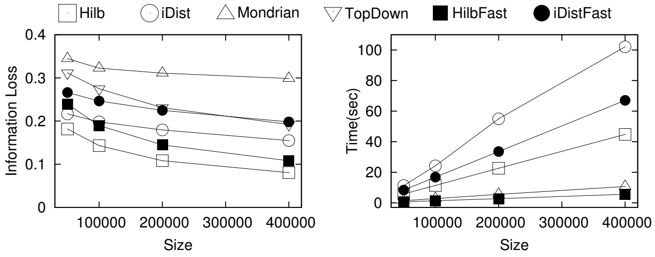


Figure 14: Census Dataset, variable size

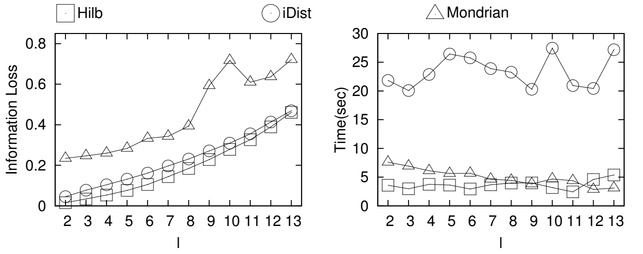


Figure 16: Census Dataset, variable ℓ

included the fast implementations of our algorithms in the graph (recall that they are approximating the partitioning cost). HilbFast is better than TopDown and Mondrian in terms of information loss. It is also very fast, achieving the same running time as Mondrian. iDistFast is similar to TopDown in terms of information loss; however it is much faster. The execution time of TopDown is around 2 hours, considerably longer than the other methods, so we do not include it in the graph.

Subsequently, for fixed $k = 50$, we vary N from $50,000$ to $400,000$ records. Figure 14 shows the results. All methods manage to reduce information loss when the size of the input increases. This is because the data density becomes higher and the probability of finding good partitions increases. Hilb and iDist are better than Mondrian and TopDown in all cases. As expected, the running time increases with the input size. Hilb needs only 40sec to anonymize 400,000 records, when $k = 50$. The execution time of TopDown (not included in the graph) is considerably more expensive: it ranges from 8min for $50,000$ records to 6 hours for $400,000$ records.

In Figure 15 we set $k = 50$, $N = 200,000$ and vary the dimensionality d of the quasi-identifier, by projecting the original 7-D data to fewer dimensions. Since Hilb and iDist are optimal for $d = 1$, for low dimensionality their information loss is close to 0 (note that the information loss of the optimal solution is typically greater than 0 due to generalization). Interestingly, for larger dimensionality, Hilb outperforms its competitors by a larger factor; therefore Hilb is suitable for real-life high-dimensional data. The running time is affected only slightly by dimensionality. Our methods face a small overhead due to the calculation of the cost of each partition in the multi-dimensional space.

6.2 ℓ-diversity Evaluation

We compare our linear 1-D heuristic ℓ-diversity algorithms (i.e., *Hilb* and *iDist*) against an ℓ-diverse variation of Mondrian, which uses the original median split heuristic, and checks for each partition whether the ℓ-diversity property is satisfied. We defer the comparison against Anatomy [20] until Section 6.3, since Anatomy does not use generalization and the GCP metric would penalize the method unfairly.

In Figure 16 we set $N = 200,000$ and vary ℓ from 2 to 13. Hilb is best in terms of information loss, followed closely by iDist. The execution time of Hilb is very low and similar to Mondrian. iDist is slower, due to the initial mapping phase.

Next (Figure 17) we set ℓ=5 and vary N from $50,000$ to $400,000$ records. As N increases, so does the data density; therefore, information loss decreases slightly for both Hilb and iDist. In terms of execution time, Hilb and Mondrian have similar performance, but Hilb is far superior in terms of information loss. Note that in all experiments the entire input table fits in the main memory. If the input table is larger than the main memory, the I/O cost of Mondrian will be much larger, since it needs to scan the input at each split. In contrast, our methods require a single scan of the input (excluding the sorting phase). Also observe that Mondrian may exhibit unpredictable, non-monotonic behavior with respect to ℓ or N. The reason is that for particular inputs, the ℓ-diversity property cannot be satisfied by any split.

In Figure 18 we set ℓ=5, $N = 200,000$ and vary the dimensionality d of Q_T. Hilb and iDist clearly outperform Mondrian. Observe that Mondrian deteriorates sharply as d increases. Also note that the execution time is virtually unaffected by dimensionality.

Finally, in Figure 19, we vary the cardinality m of the sensitive attribute (i.e., *Occupation* in our experiments). We set $N = 200,000$, and ℓ=5. In order to vary m, we aggregate continuous ranges of the sensitive attribute into a single value. All methods perform better when m increases, due to the additional freedom of choosing records with different sensitive attribute values in each partition. However, execution time is not affected by m.

6.3 Accuracy of Data Analysis Queries

In addition to the general-purpose GCP metric, in this section we employ a realistic query workload, as suggested by Ref. [11]. We compare the ℓ-diverse versions of Hilb and iDist against Anatomy and ℓ-diverse Mondrian. Anonymized data can be used to extract statistics and assist decision-making. Since these are typical OLAP operations, our workload consists of the following type of aggregation queries:

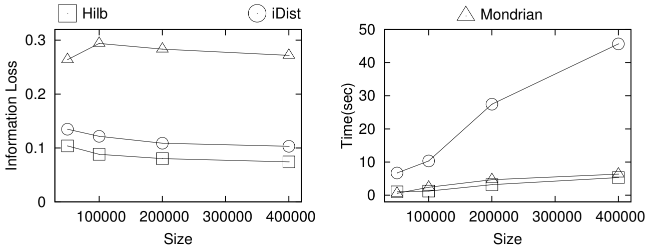


Figure 17: Census Dataset, variable size

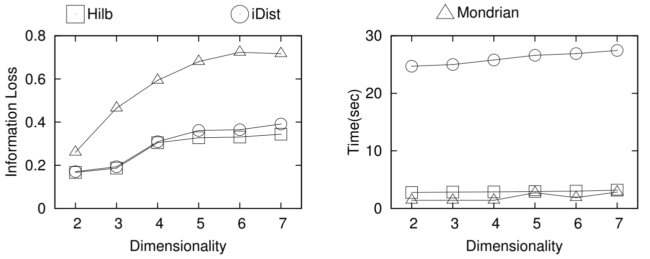


Figure 18: Census Dataset, variable dimensionality

```
SELECT QT1, QT2,..., QTi, COUNT(*)
FROM data
WHERE SA = val
GROUP BY QT1, QT2,..., QTi
```

Each QT_i is an attribute of the quasi-identifier (e.g., *Age*, *Gender*), whereas SA is a sensitive attribute (e.g., *Occupation*). The OLAP datacube [6] consists of all group-bys for all possible combinations of the quasi-identifier attributes. Interdependencies among group-by's are captured by the *datacube lattice*. Level i of the lattice corresponds to all group-bys over exactly i attributes (the higher the level, the finer the granularity of the group-by). We represent the cube as a multi-dimensional array; the cells that do not correspond to a result tuple of the above query are set to 0.

We use the CENSUS dataset and compute the entire datacube for *(i)* the original microdata (P cube) and *(ii)* the anonymized tables (Q cube). Obviously, Q is an estimation of P. Each cell of Q is computed as follows: For Anatomy, which does not use generalization, the estimation is straightforward, since the exact quasi-identifier and the probability of an SA value for a specific record, are given. On the other hand, for the generalization-based methods, we take into account the intersection of the query with each group, assuming a uniform distribution of records within the group.

Ideally, the values of all cells in cube Q should be equal to the values in the corresponding cells of P. Several methods exist to measure similarity. Ref. [20] uses the relative error: $RE = |P_C - Q_C|/P_C$, where P_C and Q_C are values of a cell in P and Q, respectively. However, this metric is undefined for $P_C = 0$. In our experiments, we use $KL\_Divergence$, which has been acknowledged as a representative metric in the data anonymization literature [8]. P and Q are modeled as multi-dimensional probability distribution functions. The estimation error is defined as:

$$KL\_Divergence(P, Q) = \sum_{\forall cell C} P_C \log \frac{P_C}{Q_C}$$

In the best case (i.e., identical P, Q), $KL\_Divergence = 0$.

In Figure 20(a), we show the query accuracy for varying ℓ at level 2 of the datacube lattice (i.e., all group-bys with two attributes); N is 200,000 records. For small ℓ, Hilb and iDist clearly outperform the competitors. Hilb is two orders of magnitude better than Mondrian, and one order of magnitude better than Anatomy, despite the fact that Anatomy is *not* using generalization but publishes the exact quasi-identifier. As ℓ increases, the extent of the anonymized groups grows proportionally in all dimensions. This is a clear disadvantage for all generalization methods; however, even for larger ℓ values, our methods outperform Mondrian by an order of magnitude, and their accuracy is only marginally worse than Anatomy.

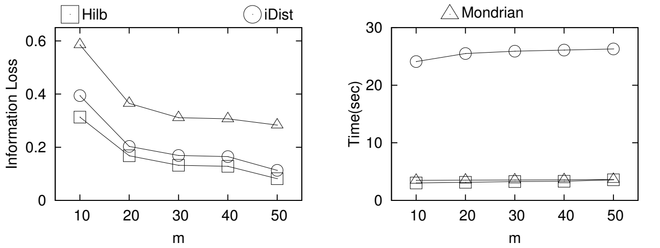


Figure 19: Variable SA Cardinality (m)

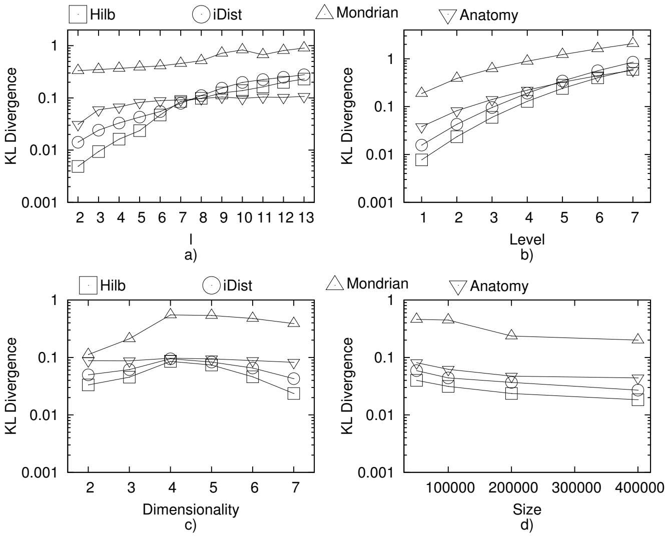


Figure 20: Query Accuracy Results

In Figure 20(b) we fix ℓ=5 and show the query accuracy for different levels of the OLAP lattice. Hilb and iDist are better than Mondrian by up to an order of magnitude, and also outperform Anatomy. Hilb and iDist are better at lower levels of the lattice (i.e., coarse-grained aggregation), since the extent of the anonymized groups is likely to be completely included in the query range. For finer granularity, Anatomy performs equally well as our methods, since it is favored by small query ranges.

In Figure 20(c) we focus on level 2 of the lattice, set ℓ=5 and vary the dimensionality d of the quasi-identifier. Lower dimensionality results to more compact ℓ-diverse groups, which improves accuracy. However, lower dimensionality also creates fine-grained group-bys, which decrease accuracy. Depending on the domains of the quasi-identifier attributes, any of the two effects may become significant. This is why there is an increasing trend until $d = 4$ and a decreasing trend afterwards. Hilb and iDist maintain an advantage over the competitors. Observe that Anatomy is not affected by dimensionality, since it does not use generalization.

Finally, in Figure 20(d) we vary the size of the input N for ℓ=5, at level 2 of the lattice. Since the extent of the queries is constant, but the density of data in the quasi-identifier space increases, accuracy increases with N.

6.4 Discussion

We demonstrated that for k-anonymity our algorithms are superior to existing techniques in terms of information loss. Hilb is the best, but is a bit slower than Mondrian. If speed is essential, HilbFast can be used. It is as fast as Mondrian and its quality is only slightly worse that Hilb.

For ℓ-diversity, Hilb is the clear winner. It is by far superior in terms of information loss and accuracy for real queries; it is also as fast as its competitors. Interestingly, Hilb outperforms Anatomy in most cases, although Anatomy implements a less secure model, by publishing the exact quasi-identifiers. This happens because Anatomy ignores the distance of the tuples in the Q_T space (see Section 2.3).

iDist also performed well, but slightly worse than Hilb. We used iDist mainly to demonstrate the versatility of our framework. For specific applications, other multi-dimensional to 1-D mappings may be more appropriate. Any such mapping can be used in our framework.

Lastly, note that our methods scale well with the input size, since the computational complexity is linear, the required memory is constant and only one scan of the data is necessary (provided the dataset is sorted).

7. CONCLUSIONS

In this paper, we developed a framework for solving the k-anonymity and ℓ-diversity problems, by mapping the multidimensional quasi-identifiers to one dimension. Both problems are NP-hard in the multidimensional space. However, we identified a set of properties for the optimal 1-D solution. Guided by these properties, we developed efficient algorithms at the 1-D space. We used two popular transformations, namely the Hilbert curve and iDistance, to solve the multidimensional problem through 1-D mapping; other transformations can easily be incorporated in our framework. The experiments demonstrate that our methods clearly outperform the existing state-of-the-art both in terms of execution time and information loss. Moreover, our algorithms are linear to the input size, therefore they are applicable to very large datasets. In the future we will study the dual problem: Given a maximum allowable information loss, identify the best possible degree of privacy (i.e., either k or ℓ).

8. REFERENCES

[1] G. Aggarwal, T. Feder, K. Kenthapadi, S. Khuller, R. Panigrahy, D. Thomas, and A. Zhu. Achieving Anonymity via Clustering. In *Proc. of ACM PODS*, pp 153–162, 2006.

[2] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas, and A. Zhu. Approximation Algorithms for k-Anonymity. *Journal of Privacy Technology*, (Paper number: 20051120001), 2005.

[3] R. J. Bayardo and R. Agrawal. Data Privacy through Optimal k-Anonymization. In *Proc. of ICDE*, pp 217–228, 2005.

[4] J. H. Friedman, J. L. Bentley, and R. A. Finkel. An Algorithm for Finding Best Matches in Logarithmic Expected Time. *ACM Transactions on Math. Softw.*, 3(3):209–226, 1977.

[5] A. Froomkin. The Death of Privacy. *Stanford Law Review*, 52(5):1461–1543, 2000.

[6] V. Harinarayan, A. Rajaraman, and J. D. Ullman. Implementing Data Cubes Efficiently. In *Proc. of ACM SIGMOD*, pp 205–216, 1996.

[7] V. S. Iyengar. Transforming Data to Satisfy Privacy Constraints. In *Proc. of SIGKDD*, pp 279–288, 2002.

[8] D. Kifer and J. Gehrke. Injecting Utility into Anonymized Datasets. In *Proc. of ACM SIGMOD*, pp 217–228, 2006.

[9] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Incognito: Efficient Full-domain k-Anonymity. In *Proc. of ACM SIGMOD*, pp 49–60, 2005.

[10] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Mondrian Multidimensional k-Anonymity. In *Proc. of ICDE*, 2006.

[11] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan. Workload-aware Anonymization. In *Proc. of KDD*, pp 277–286, 2006.

[12] N. Li, T. Li, and S. Venkatasubramanian. t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. In *Proc. of ICDE*, pp 106–115, 2007.

[13] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. l-Diversity: Privacy Beyond k-Anonymity. In *Proc. of ICDE*, 2006.

[14] A. Meyerson and R. Williams. On the Complexity of Optimal K-anonymity. In *Proc. of ACM PODS*, pp 223–228, 2004.

[15] B. Moon, H. Jagadish, and C. Faloutsos. Analysis of the Clustering Properties of the Hilbert Space-Filling Curve. *IEEE TKDE*, 13(1):124–141, 2001.

[16] P. Samarati and L. Sweeney. Generalizing Data to Provide Anonymity when Disclosing Information (abstract). In *PODS (see also Technical Report SRI-CSL-98-04)*, 1998.

[17] L. Sweeney. k-Anonymity: A Model for Protecting Privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5):557–570, 2002.

[18] Y. Tao and X. Xiao. Personalized Privacy Preservation. In *Proc. of ACM SIGMOD*, pp 229–240, 2006.

[19] R. Wong, A. Fu, J. Pei, K. Wang, S. Wan, and C. Lo. Multidimensional k-anonymization by Linear Clustering Using Space-filling Curves. TR 2006-27, Simon Fraser University, March 2006.

[20] X. Xiao and Y. Tao. Anatomy: Simple and Effective Privacy Preservation. In *Proc. of VLDB*, pp 139–150, 2006.

[21] J. Xu, W. Wang, J. Pei, X. Wang, B. Shi, and A. Fu. Utility-Based Anonymization Using Local Recoding. In *Proc. of SIGKDD*, pp 785–790, 2006.

[22] Q. Zhang, N. Koudas, D. Srivastava, and T. Yu. Aggregate Query Answering on Anonymized Tables. In *Proc. of ICDE*, pp 116–125, 2007.

[23] R. Zhang, P. Kalnis, B. C. Ooi, and K.-L. Tan. Generalized Multidimensional Data Mapping and Query Processing. *ACM TODS*, 30(3):661–697, 2005.