Mondrian Multidimensional K-Anonymity

Kristen LeFevre David J. DeWitt Raghu Ramakrishnan
University of Wisconsin, Madison

Abstract

K-Anonymity has been proposed as a mechanism for protecting privacy in microdata publishing, and numerous recoding "models" have been considered for achieving k-anonymity. This paper proposes a new multidimensional model, which provides an additional degree of flexibility not seen in previous (single-dimensional) approaches. Often this flexibility leads to higher-quality anonymizations, as measured both by general-purpose metrics and more specific notions of query answerability.

Optimal multidimensional anonymization is NP-hard (like previous optimal k-anonymity problems). However, we introduce a simple greedy approximation algorithm, and experimental results show that this greedy algorithm frequently leads to more desirable anonymizations than exhaustive optimal algorithms for two single-dimensional models.

1. Introduction

A number of organizations publish microdata for purposes such as demographic and public health research. In order to protect individual privacy, known identifiers (e.g., Name and Social Security Number) must be removed. In addition, this process must account for the possibility of combining certain other attributes with external data to uniquely identify individuals [15]. For example, an individual might be "re-identified" by joining the released data with another (public) database on Age, Sex, and Zipcode. Figure 1 shows such an attack, where Ahmed's medical information is determined by joining the released patient data with a public voter registration list.

K-anonymity has been proposed to reduce the risk of this type of attack [12, 13, 15]. The primary goal of k-anonymization is to protect the privacy of the individuals to whom the data pertains. However, subject to this constraint, it is important that the released data remain as "useful" as possible. Numerous recoding *models* have been proposed in the literature for k-anonymization [8, 9, 13, 17, 10], and often the "quality" of the published data is dictated by the model that is used. The main contributions of this paper are a new multidimensional recoding model and a greedy algorithm for k-anonymization, an approach with several important advantages:[1]

Voter Registration Data

Name	Age	Sex	Zipcode
Ahmed	25	Male	53711
Brooke	28	Female	55410
Claire	31	Female	90210
Dave	19	Male	02174
Evelyn	40	Female	02237

Patient Data

Age	Sex	Zipcode	Disease
25	Male	53711	Flu
25	Female	53712	Hepatitis
26	Male	53711	Brochitis
27	Male	53710	Broken Arm
27	Female	53712	AIDS
28	Male	53711	Hang Nail

Figure 1. Tables vulnerable to a joining attack

- The greedy algorithm is substantially *more efficient* than proposed optimal k-anonymization algorithms for single-dimensional models [2, 9, 12]. The time complexity of the greedy algorithm is $O(nlogn)$, while the optimal algorithms are exponential in the worst case.
- The greedy multidimensional algorithm often produces *higher-quality* results than optimal single-dimensional algorithms (as well as the many existing single-dimensional heuristic [6, 14, 16] and stochastic search [8, 18] algorithms).

1.1. Basic Definitions

Quasi-Identifier Attribute Set A quasi-identifer is a minimal set of attributes $X_1, ..., X_d$ in table T that can be joined with external information to re-identify individual records. We assume that the quasi-identifier is understood based on specific knowledge of the domain.

Equivalence Class A table T consists of a multiset of tuples. An equivalence class for T with respect to attributes $X_1, ..., X_d$ is the set of all tuples in T containing identical values $(x_1, ..., x_d)$ for $X_1, ..., X_d$. In SQL, this is like a GROUP BY query on $X_1, ..., X_d$.

[1] The visual representation of such recodings reminded us of the work of artist Piet Mondrian (1872-1944).

K-Anonymity Property Table T is *k-anonymous* with respect to attributes $X_1, ..., X_d$ if every unique tuple $(x_1, ..., x_d)$ in the (multiset) projection of T on $X_1, ..., X_d$ occurs at least k times. That is, the size of each equivalence class in T with respect to $X_1, ..., X_d$ is at least k.

K-Anonymization A view V of relation T is said to be a *k-anonymization* if the view modifies or generalizes the data of T according to some *model* such that V is k-anonymous with respect to the quasi-identifier.

1.2. General-Purpose Quality Metrics

There are a number of notions of microdata quality [2, 6, 8, 10, 12, 13, 14, 15, 16], but intuitively, the anonymization process should generalize or perturb the original data as little as is necessary to satisfy the k-anonymity constraint. Here we consider some simple general-purpose quality metrics, but a more targeted approach to quality measurement based on query answerability is described in Section 5.

The simplest kind of quality measure is based on the size of the equivalence classes E in V. Intuitively, the *discernability metric* (C_{DM}), described in [2], assigns to each tuple t in V a penalty, which is determined by the size of the equivalence class containing t.

$$C_{DM} = \sum_{EquivClasses\ E} |E|^2$$

As an alternative, we also propose the *normalized average equivalence class size metric* (C_{AVG}).

$$C_{AVG} = (\frac{total\_records}{total\_equiv\_classes})/(k)$$

1.3. Paper Overview

The first contribution of this paper is a new multidimensional model for k-anonymization (Section 2). Like previous optimal k-anonymity problems [1, 10], optimal multidimensional k-anonymization is NP-hard. However, for numeric data, we find that under reasonable assumptions the worst-case maximum size of equivalence classes is $O(k)$ in the multidimensional case, while in the single-dimensional model, this bound can grow linearly with the total number of records. For a simple variation of the multidimensional model, this bound is $2k$ (Section 3).

Using the multidimensional recoding model, we introduce a simple and efficient greedy algorithm that can be applied to both categorical and numeric data (Section 4). For numeric data, the results are a constant-factor approximation of optimal, as measured by the general-purpose quality metrics described in the previous section.

General-purpose quality metrics are a good starting point when the ultimate use of the published data is unknown. However, in some cases, quality might be more appropriately measured by the application consuming the published data. The second main contribution of this paper is a more sophisticated notion of quality measurement, based on a workload of aggregate queries (Section 5).

Using general-purpose metrics and a simple query workload, our experimental evaluation (Section 6) indicates that the quality of the anonymizations obtained by our greedy algorithm are often superior to those obtained by exhaustive optimal algorithms for two single-dimensional models.

The paper concludes with discussions of related and future work (Sections 7 and 8).

2. Multidimensional Global Recoding

In a relational database, each attribute has some domain of values. We use the notation D_X to denote the domain of attribute X. A *global recoding* achieves anonymity by mapping the domains of the quasi-identifier attributes to generalized or altered values [17].

Global recoding can be further broken down into two sub-classes [9]. A *single-dimensional global recoding* is defined by a function $\phi_i : D_{X_i} \rightarrow D'$ *for each attribute* X_i of the quasi-identifier. An anonymization V is obtained by applying each ϕ_i to the values of X_i in each tuple of T.

Alternatively, a *multidimensional global recoding* is defined by a *single* function $\phi : D_{X_1} \times ... \times D_{X_n} \rightarrow D'$, which is used to recode the domain of value *vectors* associated with the set of quasi-identifier attributes. Under this model, V is obtained by applying ϕ to the vector of quasi-identifier values in each tuple of T.

Multidimensional recoding can be applied to categorical data (in the presence of user-defined generalization hierarchies) or to numeric data. For numeric data, and other totally-ordered domains, (single-dimensional) "partitioning" models have been proposed [2, 8]. A *single-dimensional interval* is defined by a pair of endpoints $p, v \in D_{X_i}$ such that $p \leq v$. (The endpoints of such an interval may be open or closed to handle continuous domains.)

Single-dimensional Partitioning Assume there is a total order associated with the domain of each quasi-identifier attribute X_i. A single-dimensional partitioning defines, for each X_i, a set of *non-overlapping* single-dimensional intervals that cover D_{X_i}. ϕ_i maps each $x \in D_{X_i}$ to some *summary statistic* for the interval in which it is contained.

The released data will include simple statistics that summarize the intervals they replace. For now, we assume that these summary statistics are min-max ranges, but we discuss some other possibilities in Section 5.

This partitioning model is easily extended to multidimensional recoding. Again, assume a total order for each D_{X_i}. A *multidimensional region* is defined by a pair of d-tuples $(p_1, ..., p_d), (v_1, ..., v_d) \in D_{X_1} \times ... \times D_{X_d}$ such that $\forall i, p_i \leq v_i$. Conceptually, each region is bounded by a d-dimensional rectangular box, and each edge and vertex of this box may be either open or closed.

Age	Sex	Zipcode	Disease
[25-28]	Male	[53710-53711]	Flu
[25-28]	Female	53712	Hepatitis
[25-28]	Male	[53710-53711]	Brochitis
[25-28]	Male	[53710-53711]	Broken Arm
[25-28]	Female	53712	AIDS
[25-28]	Male	[53710-53711]	Hang Nail

Figure 2. Single-dimensional anonymization

Age	Sex	Zipcode	Disease
[25-26]	Male	53711	Flu
[25-27]	Female	53712	Hepatitis
[25-26]	Male	53711	Brochitis
[27-28]	Male	[53710-53711]	Broken Arm
[25-27]	Female	53712	AIDS
[27-28]	Male	[53710-53711]	Hang Nail

Figure 3. Multidimensional anonymization

Strict Multidimensional Partitioning A strict multidimensional partitioning defines a set of non-overlapping multidimensional regions that cover $D_{X_1} \times ... \times D_{X_d}$. ϕ maps each tuple $(x_1, ..., x_d) \in D_{X_1} \times ... \times D_{X_d}$ to a summary statistic for the region in which it is contained.

When ϕ is applied to table T (assuming each region is mapped to a unique vector of summary statistics), the tuple set in each non-empty region forms an equivalence class in V. For simplicity, we again assume that these summary statistics are ranges, and further discussion is provided in Section 5.

Sample 2-anonymizations of Patients, using single-dimensional and strict multidimensional partitioning, are shown in Figures 2 and 3. Notice that the anonymization obtained using the multidimensional model is not permissible under the single-dimensional model because the domains of Age and Zipcode are not recoded to a single set of intervals (e.g., Age 25 is mapped to either [25-26] or [25-27], depending on the values of Zipcode and Sex). However, the single-dimensional recoding is also valid under the multidimensional model.

Proposition 1 *Every single-dimensional partitioning for quasi-identifier attributes $X_1, ..., X_d$ can be expressed as a strict multidimensional partitioning. However, when $d \geq 2$ and $\forall i, |D_{X_i}| \geq 2$, there exists a strict multidimensional partitioning that cannot be expressed as a single-dimensional partitioning.*

It is intuitive that the optimal strict multidimensional partitioning must be at least as good as the optimal single-dimensional partitioning. However, the optimal k-anonymous multidimensional partitioning problem is NP-hard (Section 2.2). For this reason, we consider the worst-case upper bounds on equivalence class size for single-dimensional and multidimensional partitioning (Section 2.3).

2.1. Spatial Representation

Throughout the paper, we use a convenient spatial representation for quasi-identifiers. Consider table T with quasi-identifier attributes $X_1, ..., X_d$, and assume that there is a total ordering for each domain. The (multiset) projections of T on $X_1, ..., X_d$ can then be represented as a multiset of points in d-dimensional space. For example, Figure 4(a) shows the two-dimensional representation of Patients from Figure 1, for quasi-identifier attributes *Age* and *Zipcode*.

Similar models have been considered for rectangular partitioning in 2 dimensions [11]. In this context, the single-dimensional and multidimensional partitioning models are analogous to the "$p \times p$" and "arbitrary" classes of tilings, respectively. However, to the best of our knowledge, none of the previous optimal tiling problems have included constraints requiring minimum occupancy.

2.2. Hardness Result

There have been previous hardness results for optimal k-anonymization under the attribute- and cell-suppression models [1, 10]. The problem of *optimal strict k-anonymous multidimensional partitioning* (finding the k-anonymous strict multidimensional partitioning with the smallest C_{DM} or C_{AVG}) is also NP-hard, but this result does not follow directly from the previous results.

We formulate the following decision problem using C_{AVG}.[2] Here, a multiset of points P is equivalently represented as a set of distinct $(point, count)$ pairs.

Decisional K-Anonymous Multidimensional Partitioning Given a set P of unique *(point, count)* pairs, with points in d-dimensional space, is there a strict multidimensional partitioning for P such that for every resulting multidimensional region R_i, $\sum_{p \in R_i} count(p) \geq k$ or $\sum_{p \in R_i} count(p) = 0$, and $C_{AVG} \leq$ positive constant c?

Theorem 1 *Decisional k-anonymous multidimensional partitioning is NP-complete.*

Proof The proof is by reduction from Partition:

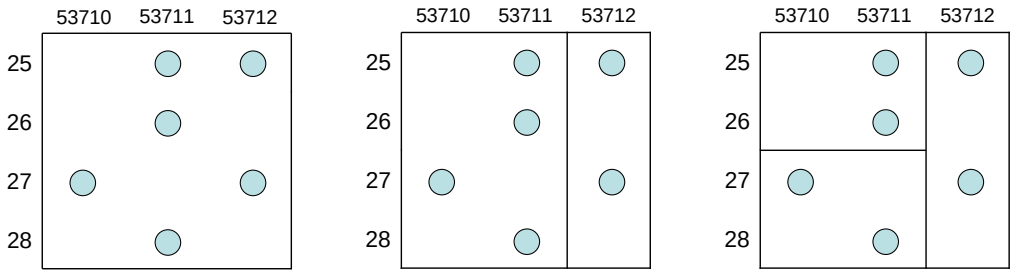
Partition Consider a set A of n positive integers $\{a_1, ..., a_n\}$. Is there some $A' \subseteq A$, such that

$$\sum_{a_i \in A'} a_i = \sum_{a_j \in A - A'} a_j \ ?$$

For each $a_i \in A$, construct a *(point, count)* pair. Let the point be defined by $(0_1, ..., 0, 1_i, 0, ..., 0_d)$ (the i^{th} coordinate is 1, and all other coordinates are 0), which resides in a d-dimensional unit-hypercube, and let the count equal a_i. Let P be the union of all such pairs.

We claim that the partition problem for A can be reduced to the following: *Let $k = \frac{\sum a_i}{2}$. Is there a k-anonymous strict multidimensional partitioning for P such*

[2] Though stated for C_{AVG}, the result is similar for C_{DM}.



(a) Patients (b) Single-Dimensional (c) Strict Multidimensional

Figure 4. Spatial representation of Patients and partitionings (quasi-identifiers Zipcode and Age)

that $C_{AVG} \leq 1$*?* To prove this claim, we show that there is a solution to the k-anonymous multidimensional partitioning problem for P if and only if there is a solution to the partition problem for A.

Suppose there exists a k-anonymous multidimensional partitioning for P. This partitioning must define two multidimensional regions, R_1 and R_2, such that $\sum_{p \in R_1} count(p) = \sum_{p \in R_2} count(p) = k = \frac{\sum a_i}{2}$, and possibly some number of empty regions. By the strictness property, these regions must not overlap. Thus, the sum of counts for the two non-empty regions constitute the sum of integers in two disjoint complementary subsets of A, and we have an equal partitioning of A.

In the other direction, suppose there is a solution to the partition problem for A. For each binary partitioning of A into disjoint complementary subsets A_1 and A_2, there is a multidimensional partitioning of P into regions $R_1, ..., R_n$ such that $\sum_{p \in R_1} count(p) = \sum_{a_i \in A_1} a_i$, $\sum_{p \in R_2} count(p) = \sum_{a_i \in A_2} a_i$, and all other R_i are empty: R_1 is defined by two points, the origin and the point p having i^{th} coordinate 1 when $a_i \in A_1$ and 0 otherwise. The bounding box for R_1 is closed at all edges and vertices. R_2 is defined by the origin and the point p having i^{th} coordinate $= 1$ when $a_i \in A_2$, and 0 otherwise. The bounding box for R_2 is open at the origin, but closed on all other edges and vertices. C_{AVG} is the average sum of counts for the non-empty regions, divided by k. In this construction, $C_{AVG} = 1$, and $R_1, ..., R_n$ is a k-anonymous multidimensional partitioning of P.

Finally, a given solution to the decisional k-anonymous multidimensional partitioning problem can be verified in polynomial time by scanning the input set of $(point, count)$ pairs, and maintaining a sum for each region.□

2.3. Bounds on Partition Size

It is also interesting to consider worst-case upper bounds on the size of partitions resulting from single-dimensional and multidimensional partitioning. This section presents two results, the first of which indicates that for a constant-sized quasi-identifier, this upper bound depends only on k and the maximum number of duplicate copies of a single point (Theorem 2). This is in contrast to the second result (Theorem 3), which indicates that for single-dimensional partitioning, this bound may grow linearly with the total number of points.

In order to state these results, we first define some terminology. A *multidimensional cut* for a multiset of points is an axis-parallel binary cut producing two disjoint multisets of points. Intuitively, such a cut is allowable if it does not cause a violation of k-anonymity.

Allowable Multidimensional Cut Consider multiset P of points in d-dimensional space. A cut perpendicular to axis X_i at x_i is allowable if and only if $Count(P.X_i > x_i) \geq k$ and $Count(P.X_i \leq x_i) \geq k$.

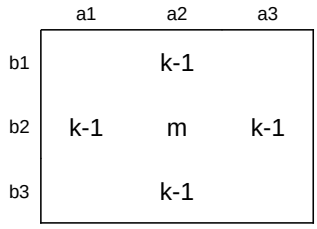
A *single-dimensional cut* is also axis-parallel, but considers all regions in the space to determine allowability.

Allowable Single-Dimensional Cut Consider a multiset P of points in d-dimensional space, and suppose we have already made S single-dimensional cuts, thereby separating the space into disjoint regions $R_1, ..., R_m$. A single-dimensional cut perpendicular to X_i at x_i is allowable, given S, if $\forall R_j$ overlapping line $X_i = x_i$, $Count(R_j.X_i > x_i) \geq k$ and $Count(R_j.X_i \leq x_i) \geq k$.

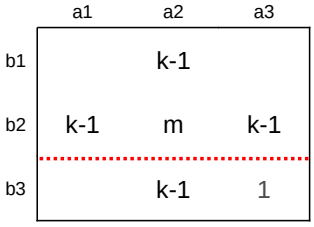
Notice that recursive allowable multidimensional cuts will result in a k-anonymous strict multidimensional partitioning for P (although not all strict multidimensional partitionings can be obtained in this way), and a k-anonymous single-dimensional partitioning for P is obtained through successive allowable single-dimensional cuts.

For example, in Figures 4(b) and (c), the first cut occurs on the *Zipcode* dimension at 53711. In the multidimensional case, the left-hand side is cut again on the *Age* dimension, which is allowable because it does not produce a region containing fewer than k points. In the single-dimensional case, however, once the first cut is made, there are no remaining allowable single-dimensional cuts. (Any cut perpendicular to the Age axis would result in a region on the right containing fewer than k points.)

Intuitively, a partitioning is considered minimal when there are no remaining allowable cuts.



(a) A set of points for which there is no allowable cut



(b) Adding a single point produces an allowable cut

Figure 5. 2-Dimensional example of equivalence class size bound

Minimal Strict Multidimensional Partitioning Let $R_1, ..., R_n$ denote a set of regions induced by a strict multidimensional partitioning, and let each region R_i contain multiset P_i of points. This multidimensional partitioning is minimal if $\forall i$, $|P_i| \geq k$ and there exists no allowable multidimensional cut for P_i.

Minimal Single-Dimensional Partitioning A set S of allowable single-dimensional cuts is a minimal single-dimensional partitioning for multiset P of points if there does not exist an allowable single-dimensional cut for P given S.

The following two theorems give upper-bounds on partition size for minimal multidimensional and single-dimensional partitioning, respectively.

Theorem 2 *If $R_1, ..., R_n$ denote the set of regions induced by a minimal strict multidimensional partitioning for multiset of points P, the maximum number of points contained in any R_i is $2d(k-1)+m$, where m is the maximum number of copies of any distinct point in P.*

Proof The proof has two parts. First, we show that there exists a multiset P of points in d-dimensional space such that $|P| = 2d(k-1)+m$ and that there is no allowable multidimensional cut for P. Let $\widehat{x}_i$ denote some value on axis X_i such that $\widehat{x}_i - 1$ and $\widehat{x}_i + 1$ are also values on axis X_i, and let P initially contain m copies of the point $(\widehat{x}_1, \widehat{x}_2, ..., \widehat{x}_d)$. Add to P $k-1$ copies of each of the following points:

$(\widehat{x}_1 - 1, \widehat{x}_2, ..., \widehat{x}_d)$, $(\widehat{x}_1 + 1, \widehat{x}_2, ..., \widehat{x}_d)$,
$(\widehat{x}_1, \widehat{x}_2 - 1, ..., \widehat{x}_d)$, $(\widehat{x}_1, \widehat{x}_2 + 1, ..., \widehat{x}_d)$,
...
$(\widehat{x}_1, \widehat{x}_2, ..., \widehat{x}_d - 1)$, $(\widehat{x}_1, \widehat{x}_2, ..., \widehat{x}_d + 1)$

For example, Figure 5 shows P in 2 dimensions. By addition, $|P| = 2d(k-1)+m$, and by projecting P onto any X_i we obtain the following point counts:

$$Count(X_i) = \begin{cases} k-1, & X_i = \widehat{x}_i - 1 \\ m + 2(d-1)(k-1), & X_i = \widehat{x}_i \\ k-1, & X_i = \widehat{x}_i + 1 \\ 0, & \text{otherwise} \end{cases}$$

Based on these counts, it is clear that any binary cut perpendicular to axis X_i would result in some partition containing fewer than k points.

Second, we show that for any multiset of points P in d-dimensional space such that $|P| > 2d(k-1)+m$, there exists an allowable multidimensional cut for P.

Consider some P in d-dimensional space, such that $|P| = 2d(k-1)+m+1$, and let $\widehat{x}_i$ denote the median value of P projected on axis X_i. If there is no allowable cut for P, we claim that there are at least $m+1$ copies of point $(\widehat{x}_1, ..., \widehat{x}_d)$ in P, contradicting the definition of m.

For every dimension $i = 1, ..., d$, if there is no allowable cut perpendicular to axis X_i, then (because $\widehat{x}_i$ is the median) $Count(X_i < \widehat{x}_i) \leq k-1$ and $Count(X_i > \widehat{x}_i) \leq k-1$. This means that $Count(X_i = \widehat{x}_i) \geq 2(d-1)(k-1)+m+1$. Thus, over d dimensions, we find $Count(X_1 = \widehat{x}_1 \wedge ... \wedge X_d = \widehat{x}_d) \geq m+1$. □

Theorem 3 *The maximum number of points contained in any region R resulting from a minimal single-dimensional partitioning of a multiset of points P in d-dimensional space ($d \geq 2$) is $O(|P|)$.*

Proof We construct a multiset of points P, and a minimal single-dimensional partitioning for P, such that the greatest number of points in a resulting region is $O(|P|)$.

Consider a quasi-identifier attribute X with domain D_X, and a finite set $V_X \subseteq D_X$ with a point $\widehat{x} \in V_X$.

Initially, let P contain precisely $2k-1$ points p having $p.X = \widehat{x}$. Then add to P an arbitrarily large number of points q, each with $q.X \in V_X$, but $q.X \neq \widehat{x}$, and such that for each $v \in V_X$ there are at least k points in the resulting set P having $X = v$.

By construction, if $|V_x| = r$, there are $r-1$ allowable single-dimensional cuts for P perpendicular to X (at each point in V_X), and we denote this set of cuts S. However, there are no allowable single-dimensional cuts for P given S (perpendicular to any other axis). Thus, S is a minimal single-dimensional partitioning, and the size of the largest resulting region is $O(|P|)$. □

3. Multidimensional Local Recoding

In contrast to global recoding, *local recoding* models map (non-distinct) individual data items to generalized values

[17]. Formally, a local recoding function, which we will denote ϕ^* to distinguish it from global recoding functions, maps each (non-distinct) tuple $t \in T$ to some recoded tuple t'. V is obtained by replacing each tuple $t \in T$ with $\phi^*(t)$. Several local recoding models have been considered in the literature, some of which are outlined in [9]. In this section, we describe one such model that relaxes the requirements of strict multidimensional partitioning.

Relaxed Multidimensional Partitioning A relaxed multidimensional partitioning for relation T defines a set of (potentially *overlapping*) distinct multidimensional regions that cover $D_{X_1} \times ... \times D_{X_d}$. Local recoding function ϕ^* maps each tuple $(x_1, ..., x_d) \in T$ to a summary statistic for *one* of the regions in which it is contained.

This relaxation offers an extra degree of flexibility. For example, consider generating a 3-anonymization of Patients, and suppose *Zipcode* is the single quasi-identifier attribute. Using the strict model, we would need to recode the Zipcode value in each tuple to [53710-53712]. Under the relaxed model, this recoding can be performed on a tuple-by-tuple basis, mapping two occurrences of 53711 to $[53710 - 53711]$ and one occurrence to $[53711 - 53712]$.

Proposition 2 *Every strict multidimensional partitioning can be expressed as a relaxed multidimensional partitioning. However, if there are at least two tuples in table T having the same vector of quasi-identifier values, there exists a relaxed multidimensional partitioning that cannot be expressed as a strict multidimensional partitioning.*

Under the relaxed model, a partitioning is not necessarily defined by binary cuts. Instead, a set of points is partitioned by defining two (possibly overlapping) multidimensional regions P_1 and P_2, and then mapping each point to either P_1 or P_2 (but not both). In this case, the upper-bound on the size of a minimal partition (one that cannot be divided without violating k-anonymity) is $2k - 1$.

4. A Greedy Partitioning Algorithm

Using multidimensional partitioning, a k-anonymization is generated in two steps. In the first step, multidimensional regions are defined that cover the domain space, and in the second step, recoding functions are constructed using summary statistics from each region. In the previous sections, we alluded to a recursive algorithm for the first step. In this section we outline a simple scalable algorithm, reminiscent of those used to construct kd-trees [5], that can be adapted to either strict or relaxed partitioning. The second step is described in more detail in Section 5

The strict partitioning algorithm is shown in Figure 6. Each iteration must choose the dimension and value about which to partition. In the literature about kd-trees, one strategy used for obtaining uniform occupancy was median-partitioning [5]. In Figure 6, the split value is the median of $partition$ projected on dim. Like kd-tree construction, the time complexity is $O(nlogn)$, where $n = |T|$.

```
Anonymize(partition)
 if (no allowable multidimensional cut for partition)
  return φ : partition → summary
 else
  dim ← choose_dimension()
  fs ← frequency_set(partition, dim)
  splitVal ← find_median(fs)
  lhs ← {t ∈ partition : t.dim ≤ splitVal}
  rhs ← {t ∈ partition : t.dim > splitVal}
  return Anonymize(rhs) ∪ Anonymize(lhs)
```

Figure 6. Top-down greedy algorithm for strict multidimensional partitioning

If there exists an allowable multidimensional cut for partition P perpendicular to some axis X_i, then the cut perpendicular to X_i at the median is allowable. By Theorem 2, the greedy (strict) median-partitioning algorithm results in a set of multidimensional regions, each containing between k and $2d(k-1)+m$ points, where m is the maximum number of copies of any distinct point.

We have some flexibility in choosing the dimension on which to partition. As long as we make an allowable cut when one exists, this choice does not affect the partition-size upper-bound. One heuristic, used in our implementation, chooses the dimension with the widest (normalized) range of values [5]. Alternatively, it may be possible to choose a dimension based on an anticipated workload.

The partitioning algorithm in Figure 6 is easily adapted for relaxed partitioning. Specifically, the points falling at the median (where $t.dim = splitVal$) are divided evenly between lhs_child and rhs_child such that $|lhs\_child| = |rhs\_child|$ ($+1$ when $|partition|$ is odd). In this case, there is a $2k - 1$ upper-bound on partition size.

Finally, a similar greedy multidimensional partitioning strategy can be used for categorical attributes in the presence of user-defined generalization hierarchies. However, our quality upper-bounds do not hold in this case.

4.1. Bounds on Quality

Using our upper bounds on partition size, it is easy to compute bounds for the general-purpose metrics described in Section 1.2 and totally-ordered attributes. By definition, k-anonymity requires that every equivalence class contain at least k records. For this reason, the optimal achievable value of C_{DM} (denoted $C_{DM}*$) $\geq k * total\_records$, and $C_{AVG}* \geq 1$.

For strict multidimensional partitioning, assume that the points in each distinct partition are mapped to a unique vector of summary statistics. We showed that under the

greedy algorithm, for each equivalence class E, $|E| \leq 2d(k-1)+m$, where m is the maximum number of copies of any distinct quasi-identifier tuple.

C_{DM} is maximized when the tuples are divided into the largest possible equivalence classes, so $C_{DM} \leq (2d(k-1)+m) * total\_records$. Thus,

$$\frac{C_{DM}}{C_{DM*}} \leq \frac{2d(k-1)+m}{k}$$

Similarly, C_{AVG} is maximized when the tuples are divided into the largest possible equivalence classes, so $C_{AVG} \leq (2d(k-1)+m)/k$.

$$\frac{C_{AVG}}{C_{AVG*}} \leq \frac{2d(k-1)+m}{k}$$

Observe that for constant d and m, the strict greedy algorithm is a constant-factor approximation, as measured by C_{DM} and C_{AVG}. If d varies, but m/k is constant, the approximation is $O(d)$.

For relaxed multidimensional partitioning, the greedy algorithm produces a 2-approximation because $C_{DM} \leq 2k * total\_records$, and $C_{AVG} \leq 2$.

4.2. Scalability

When the table T to be anonymized is larger than the available memory, the main scalability issue to be addressed is finding the median value of a selected attribute within a given partition.

We propose a solution to this problem based on the idea of a *frequency set*. The frequency set of attribute A for partition P is the set of unique values of A in P, each paired with an integer indicating the number of times it appears in P. Given the frequency set of A for P, the median value is found using a standard median-finding algorithm.

Because individual frequency sets contain just one entry per value in the domain of a particular attribute, and are much smaller than the size of the data itself, it is reasonable to assume that a single frequency set will fit in memory. For this reason, in the worst case, we must sequentially scan the database at most twice, and write once, per level of the recursive partitioning "tree." The data is first scanned once to find the median, and then scanned and written once to re-partition the data into two "runs" (lhs and rhs) on disk.

It is worth noting that this scheme may be further optimized to take advantage of available memory because, in practice, the frequency sets for multiple attributes may fit in memory. One approach would be similar to the scalable algorithms used for decision tree construction in [7].

5. Workload-Driven Quality Measurement

The general-purpose quality metrics in Section 1.2 are a good place to start when the application that ultimately consumes the anonymized data is unknown. However, in some cases, the publisher may want to consider an anticipated workload, such as building a data mining model [6, 8, 16], or answering a set of aggregate queries. This section introduces the latter problem, including examples where multidimensional recoding provides needed flexibility.

Consider a set of queries with selection predicates (equality or range) of the form `attribute <oper> constant` and an aggregate function (COUNT, SUM, AVG, MIN, and MAX). Our ability to answer this type of queries from anonymized data depends on two main factors: the type of *summary statistic(s)* released for each attribute, and the degree to which the selection predicates in the workload *match* the range boundaries in the anonymous data.

The choice of summary statistics influences our ability to compute various aggregate functions.[3] In this paper, we consider releasing two summary statistics for each attribute A and equivalence class E:

- **Range statistic (R)** So far, all of our examples have involved a single summary statistic defined by the range of values for A appearing in E, which allows for easy computation of MIN and MAX aggregates.
- **Mean Statistic (M)** We also consider a summary statistic defined by the mean value of A appearing in E, which allows for the computation of AVG and SUM.

When choosing summary statistics, it is important to consider potential avenues for inference. Notice that in some cases simply releasing the minimum-maximum range allows for some inferences about the distribution of values within an equivalence class. For example, consider an attribute A, and let $k = 2$. Suppose that an equivalence class of the released anonymization contains two tuples, and A is summarized by the range $[0-1]$. It is easy to infer that in one of the original tuples $A = 0$, and in the other $A = 1$.

This type of inference about distribution (which may also arise in single-dimensional recoding) is not likely to pose a problem in preventing joining attacks because, without background knowledge, it is still impossible for an adversary to distinguish the tuples within an equivalence class from one another.

The second factor influencing our ability to answer aggregate queries is the degree to which the selection predicates in the given workload "match" the boundaries of the range statistics in the released anonymization. In many ways, this is analogous to matching indices and selection predicates in traditional query processing.

Predicate-Range Matching A selection predicate $Pred$ conceptually divides the original table T into two sets of tuples, T^T_{Pred} and T^F_{Pred} (those that satisfy the predicate and

[3] Certain types of aggregate functions (e.g., MEDIAN) are ill-suited to this type of computation. We do not know of any way to compute such functions from these summary statistics.

those that do not). When range statistics are published, we say that an anonymization V *matches* a boolean predicate $Pred$ if every tuple $t \in T^T_{Pred}$ is mapped to an equivalence class in V containing no tuples from T^F_{Pred}.

To illustrate these ideas, consider a workload containing two queries:

```
SELECT AVG(Age)
FROM Patients
WHERE Sex = 'Male'
```

```
SELECT COUNT(*)
FROM Patients
WHERE Sex = 'Male'
AND Age ≤ 26
```

A strict multidimensional anonymization of Patients is given in Figure 7, including two summary statistics (range and mean) for Age. Notice that the mean allows us to answer the first query precisely and accurately. The second query can also be answered precisely because the predicate matches a single equivalence class in the anonymization. Comparing this with the single-dimensional recoding shown in Figure 2, notice that it would be impossible to answer the second query precisely using the single-dimensional recoding.

When a workload consists of many queries, even a multidimensional anonymization might not match every selection predicate. An exhaustive discussion of query processing over imprecise data is beyond the scope of this paper. However, when no additional distribution information is available, a simple approach assumes a uniform distribution of values for each attribute within each equivalence class. The effects of multidimensional versus single-dimensional recoding, with respect to a specific query workload, are explored empirically in Section 6.3.

Our work on workload-driven anonymization is preliminary, and in this paper, the workload is primarily used as an evaluation tool. One of the most important future directions is directly integrating knowledge of an anticipated workload into the anonymization algorithm. Formally, a query workload can be expressed as a set of *(multdimensional region, aggregate, weight)* triples, where the boundaries of each region are determined by the selection predicates in the workload. Each query is also assigned a weight indicating its importance with respect to the rest of the workload. When a selection predicate in the workload does not exactly match the boundaries of one or more equivalence classes, evaluating this query over the anonymized data will incur some *error*. This error can be defined as the normalized difference between the result of evaluating the query on the anonymous data, and the result on the original data. Intuitively, the task of a workload-driven algorithm is to generate an anonymization that minimizes the weighted sum of such errors.

6. Experimental Evaluation

Our experiments evaluated the quality of anonymizations produced by our greedy multidimensional algorithm by comparing these results with those produced by *optimal* algorithms for two other models: full-domain generalization [9, 12], and single-dimensional partitioning [2, 8]. The specific algorithms used in the comparison (Incognito [9] and K-Optimize [2]) were chosen for efficiency, but any exhaustive algorithm for these models would yield the same result. It is important to note that the exhaustive algorithms are exponential in the worst case, and they run many times slower than our greedy algorithm. Nonetheless, the quality of the results obtained by the latter is often superior.

Age(R)	Age(M)	Sex(R)	Zipcode(R)	Disease
$[25-26]$	25.5	Male	53711	Flu
$[25-27]$	26	Female	53712	Hepatitis
$[25-26]$	25.5	Male	53711	Brochitis
$[27-28]$	27.5	Male	$[53710-53711]$	Broken Arm
$[25-27]$	26	Female	53712	AIDS
$[27-28]$	27.5	Male	$[53710-53711]$	Hang Nail

Figure 7. A 2-anonymization with multiple summary statistics

Distribution	(Discrete Uniform, Discrete Normal)
Attributes	Total quasi-identifier attributes
Cardinality	Distinct values per attribute
Tuples	Total tuples in table
Std. Dev. (σ)	With respect to standard normal (Normal only)
Mean (μ)	(Normal only)

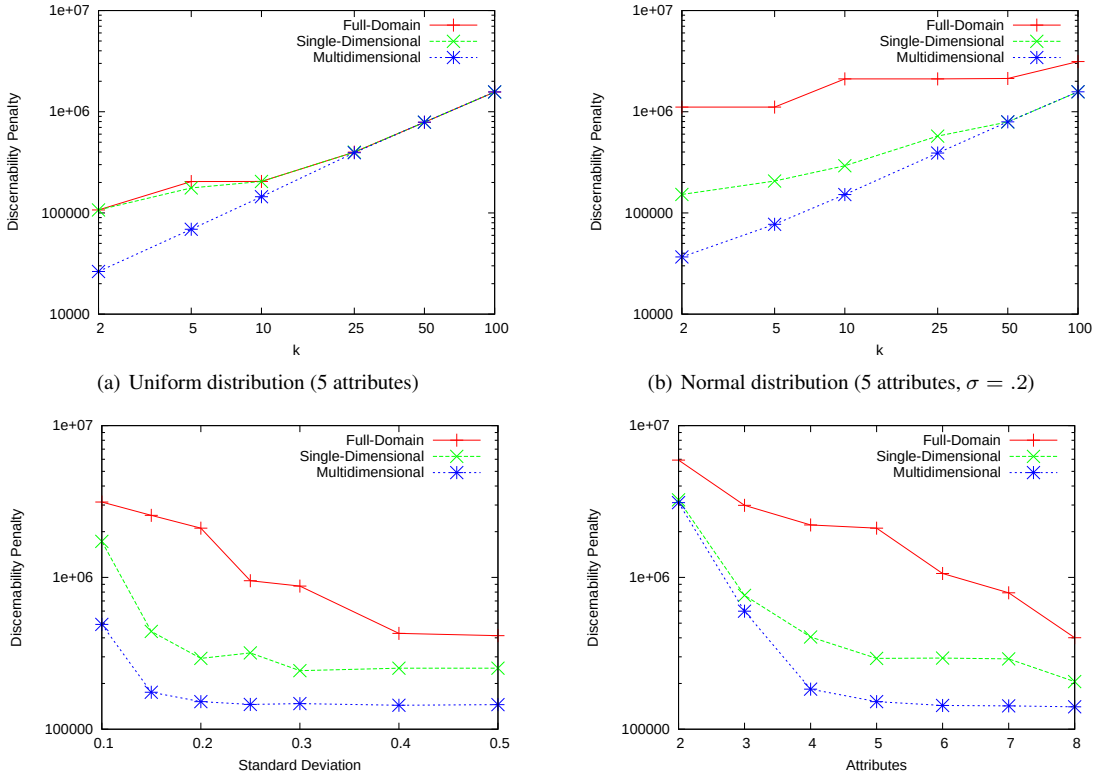
Figure 8. Parameters of synthetic generator

For these experiments, we used both synthetic and real-world data. We compared quality, using general-purpose metrics, and also with respect to a simple query workload.

6.1. Experimental Data

For some experiments, we used a synthetic data generator, which produced two discrete joint distributions: *discrete uniform* and *discrete normal*. We limited the evaluation to discrete domains so that the exhaustive algorithms would be tractable without pre-generalizing the data. To generate the discrete normal distribution, we first generated the multivariate normal distribution, and then discretized the values of each attribute into equal-width ranges. The parameters are described in Figure 8.

In addition to synthetic data, we also used the Adults database from the UC Irvine Machine Learning Repository [3], which contains census data, and has become a de facto benchmark for k-anonymity. We configured this data set as it was configured for the experiments reported in [2], using eight regular attributes, and removing tuples with missing values. The resulting database contained 30,162 records. For the partitioning experiments, we imposed an intuitive ordering on each attribute, but unlike [2], we eliminated all hierarchical constraints for both models. For the full-domain experiments, we used the same generalization hierarchies that were used in [9].



(a) Uniform distribution (5 attributes)

(b) Normal distribution (5 attributes, $\sigma = .2$)

(c) Normal distribution (5 attributes, $k = 10$)

(d) Normal distribution ($k = 10, \sigma = .2$)

Figure 9. Quality comparisons for synthetic data using the discernability metric

6.2. General-Purpose Metrics

We first performed some experiments comparing quality using general-purpose metrics. Here we report results for the discernability penalty (Section 1.2), but the comparisons are similar for the average equivalence class size.

The first experiment compared the three models for varied k. We fixed the number of tuples at 10,000, the per-attribute cardinality at 8, and the number of attributes at 5. For the full-domain generalization model, we constructed generalization hierarchies using binary trees. The results for the uniform distribution are shown in Figure 9(a). Results for the discrete normal distribution ($\mu = 3.5$, $\sigma = .2$) are given in Figure 9(b). We found that greedy multidimensional partitioning produced "better" generalizations than the other algorithms in both cases. However, the magnitude of this difference was much more pronounced for the non-uniform distribution.

Following this observation, the second experiment compared quality using the same three models, but varied the standard deviation (σ) of the synthetic data. (Small values of σ indicate a high degree of non-uniformity.) The number of attributes was again fixed at 5, and k was fixed at 10. The results (Figure 9(c)) show that the difference in quality is most pronounced for non-uniform distributions.

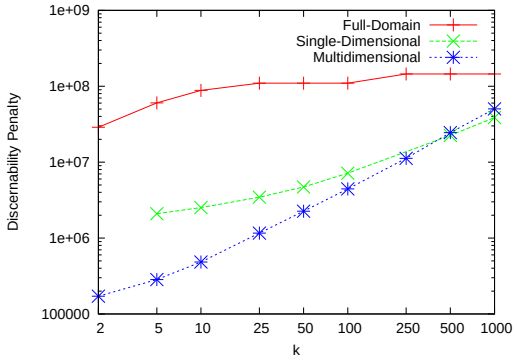
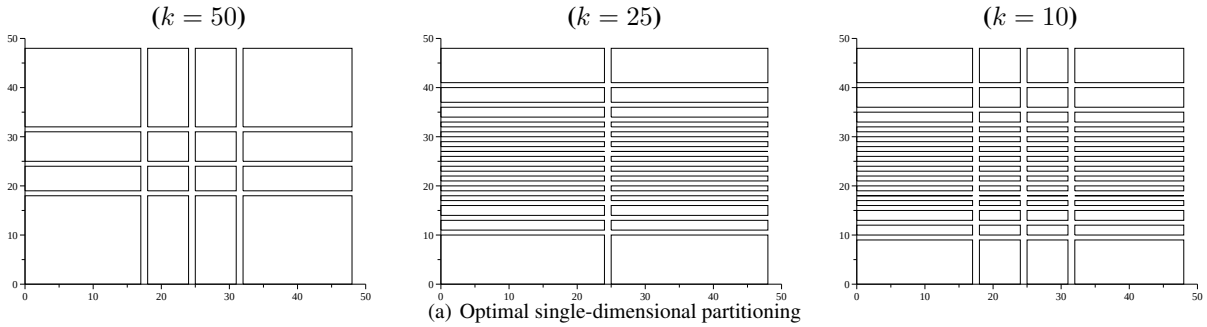


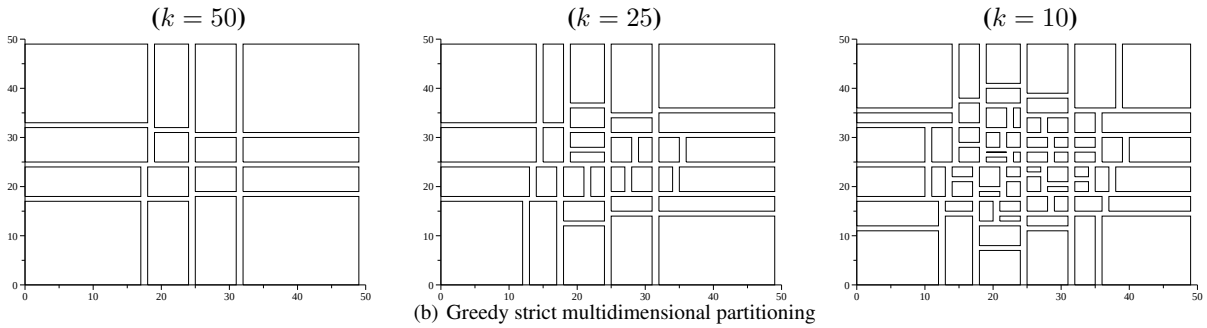
Figure 10. Quality comparison for Adults database using discernability metric

The next experiment measured quality for varied quasi-identifier size, with $\sigma = .2$ and $k = 10$. As the number of attributes increased, the observed discernability penalty decreased for each of the three models (Figure 9(d)). At first glance, this result is counter-intuitive. However, this decrease is due to the sparsity of the original data, which contains fewer duplicate tuples as the number of attributes increases.

In addition to the synthetic data, we compared the three algorithms using the Adults database (Figure 10). Again, we found that greedy multidimensional partitioning gener-



(a) Optimal single-dimensional partitioning



(b) Greedy strict multidimensional partitioning

Figure 11. Anonymizations for two attributes with a discrete normal distribution ($\mu = 25$, $\sigma = .2$).

Predicate on X

k	Model	Mean Error	Std. Dev.
10	Single	7.73	5.94
10	Multi	4.66	3.26
25	Single	12.68	7.17
25	Multi	5.69	3.86
50	Single	7.73	5.94
50	Multi	7.94	5.87

Predicate on Y

k	Model	Mean Error	Std. Dev.
10	Single	3.18	2.56
10	Multi	4.03	3.44
25	Single	5.06	4.17
25	Multi	5.67	3.80
50	Single	8.25	6.15
50	Multi	8.06	5.58

Figure 12. Error for count queries with single-attribute selection predicates

ally produced the best results.

6.3. Workload-Based Quality

We also compared the optimal single-dimensional and greedy multidimensional partitioning algorithms with respect to a simple query workload, using a synthetic data set containing 1000 tuples, with two quasi-identifier attributes (discrete normal, each with cardinality 50, $\mu = 25$, $\sigma = .2$). Visual representations of the resulting partitionings are given in Figures 11(b) and 11(a).

Multidimensional partitioning does an excellent job at capturing the underlying multivariate distribution. In contrast, we observed that for non-uniform data and small k, single-dimensional partitioning tends to reflect the distribution of just one attribute. However, the optimal single-dimensional anonymization is quite sensitive to the underlying data, and a small change to the synthetic data set often dramatically changes the resulting anonymization.

This tendency to "linearize" attributes has an impact on query processing over the anonymized data. Consider a simple workload for this two-attribute data set, consisting of queries of the form "SELECT COUNT(*) WHERE $\{X, Y\}$ = $value$", where X and Y are the quasi-identifier attributes, and $value$ is an integer between 0 and 49. (In Figures 11(a) and 11(b), X and Y are displayed on the horizontal and vertical axes.) We evaluated the set of queries of this form over each anonymization and the original data set. When a predicate did not match any partition, we assumed a uniform distribution within each partition.

For each anonymization, we computed the mean and standard deviation of the absolute error over the set of queries in the workload. These results are presented in Figure 12. As is apparent from Figures 11(a) and 11(b), and from the error measurements, queries with predicates on Y are more accurately answered from the single-dimensional anonymization than are queries with predicates on X. The observed error is more consistent across queries using the multidimensional anonymization.

7. Related Work

Many recoding models have been proposed in the literature for guaranteeing k-anonymity. The majority of these

models have involved user-defined value generalization hierarchies [6, 8, 9, 12, 14, 16]. Recently, partitioning models have been proposed to automatically produce generalization hierarchies for totally-ordered domains [2, 8]. However, these recoding techniques have all been single-dimensional.

In addition to global recoding models, simpler local recoding models have also been considered, including suppressing individual data cells. Several approximation algorithms have been proposed for the problem of finding the k-anonymization that suppresses the fewest cells [1, 10].

In another related direction, Chawla et al. [4] propose a theoretical framework for privacy in data publishing based on private histograms. This work describes a recursive sanitization algorithm in multidimensional space. However, in their problem formulation, minimum partition-occupancy is not considered to be an absolute constraint.

Finally, several papers have considered evaluating the results of k-anonymization algorithms based on a particular data mining task, such as building a decision tree [6, 8, 16], but quality evaluation based on a query workload has not previously been explored.

8. Conclusion and Future Work

In this paper, we introduced a multidimensional recoding model for k-anonymity. Although optimal multidimensional partitioning is NP-hard, we provide a simple and efficient greedy approximation algorithm for several general-purpose quality metrics. An experimental evaluation indicates that often the results of this algorithm are actually *better* than those produced by more expensive optimal algorithms using other recoding models.

The second main contribution of this paper is a more targeted notion of quality measurement, based on a workload of aggregate queries. The second part of our experimental evaluation indicated that, for workloads involving predicates on multiple attributes, the multidimensional recoding model often produces more desirable results.

There are a number of promising areas for future work. In particular, as mentioned in Section 5, we are considering ways of integrating an anticipated query workload directly into the anonymization algorithms. Also, we suspect that multidimensional recoding would lend itself to creating anonymizations that are useful for building data mining models since the partitioning pattern more faithfully reflects the multivariate distribution of the original data.

9. Acknowledgements

Our thanks to Vladlen Koltun for pointing us toward the median-partitioning strategy for *kd*-trees, to Roberto Bayardo for providing the unconstrained single-dimensional quality results for the Adults database, to Miron Livny for his ideas about the synthetic data generator, and to three anonymous reviewers for their helpful feedback.

This work was supported by an IBM Ph.D. Fellowship and NSF Grant IIS-0524671.

References

[1] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas, and A. Zhu. Anonymizing tables. In *ICDT*, 2005.

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

[3] C. Blake and C. Merz. UCI repository of machine learning databases, 1998.

[4] S. Chawla, C. Dwork, F. McSherry, A. Smith, and H. Wee. Toward privacy in public databases. In *2nd Theory of Cryptography Conference*, 2005.

[5] J. Friedman, J. Bentley, and R. Finkel. An algorithm for finding best matches in logarithmic time. *ACM Trans. on Mathematical Software*, 3(3), 1977.

[6] B. Fung, K. Wang, and P. Yu. Top-down specialization for information and privacy preservation. In *ICDE*, 2005.

[7] J. Gehrke, R. Ramakrishnan, and V. Ganti. RainForest: A framework for fast decision tree construction of large datasets. In *VLDB*, 1998.

[8] V. Iyengar. Transforming data to satisfy privacy constraints. In *ACM SIGKDD*, 2002.

[9] K. LeFevre, D.DeWitt, and R. Ramakrishnan. Incognito: Efficient full-domain k-anonymity. In *ACM SIGMOD*, 2005.

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

[11] S. Muthakrishnan, V. Poosala, and T. Suel. On rectangular partitionings in two dimensions: Algorithms, complexity, and applications. In *ICDT*, 1998.

[12] P. Samarati. Protecting respondents' identities in microdata release. *IEEE Trans. on Knowledge and Data Engineering*, 13(6), 2001.

[13] P. Samarati and L. Sweeney. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. Technical Report SRI-CSL-98-04, SRI Computer Science Laboratory, 1998.

[14] L. Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *Int'l Journal on Uncertainty, Fuzziness, and Knowledge-based Systems*, 10(5):571–588, 2002.

[15] L. Sweeney. K-anonymity: A model for protecting privacy. *Int'l Journal on Uncertainty, Fuzziness, and Knowledge-based Systems*, 10(5):557–570, 2002.

[16] K. Wang, P. Yu, and S. Chakraborty. Bottom-up generalization: A data mining solution to privacy protection. In *ICDM*, 2004.

[17] L. Willenborg and T. deWaal. *Elements of Statistical Disclosure Control*. Springer Verlag, 2000.

[18] W. Winkler. Using simulated annealing for k-anonymity. Research Report 2002-07, US Census Bureau Statistical Research Division, 2002.