Privacy by Fake Data: A Geometric Approach

Victor Alvarez[*] Erin Chambers[†] László Kozma[‡]

Abstract

We study the following algorithmic problem: given n points within a finite d-dimensional box, what is the smallest number of extra points that need to be added to ensure that every d-dimensional unit box is either empty, or contains at least k points. We motivate the problem through an application to data privacy, namely k-anonymity. We show that minimizing the number of extra points to be added is strongly NP-complete, but admits a Polynomial Time Approximation Scheme (PTAS). In some sense, this is the best we can hope for, since a Fully Polynomial Time Approximation Scheme (FPTAS) is not possible, unless P=NP.

1 Introduction

Data privacy is a fundamental problem associated to data mining. On one hand, we would like to make data publicly available so that data mining or analysis is possible. On the other hand, we would like to make sure that the identity of an individual is not disclosed and no extra information is revealed as a result of mining. Several approaches have been proposed for alleviating the inherent tension between the two goals. Two of the more popular frameworks are k-anonymity [3] and differential privacy [7].

In differential privacy, one controls the way a database is accessed, and adds noise to the results of queries to the database. The idea is essentially to ensure that the results of any query, or analysis, with or without the data of one individual have similar distributions.

The idea behind k-anonymity is to ensure that for every query there are at least k records that are indistinguishable from each other. This is usually achieved by suppression or generalization, i.e., selective deletion of parts of data - which hopefully does not substantially affect the results of analysis using the data. The larger the value of k, the greater the extent of privacy. Meyerson and Williams [5] have studied the complexity of computing the minimum amount of generalization and suppression necessary for k-anonymity, and proved that it is an NP-hard problem. They also give an $O(k \log k)$ approximation algorithm. LeFevre, DeWitt and Ramakrishnan [6] studied the optimization problem in a multidimensional model. They proved NP-hardness and gave a greedy algorithm that seems to perform well in practice.

In this paper we concentrate on achieving k-anonymity, assuming that the queries are sufficiently *broad*. This condition describes a situation in which an adversary has only partial or inaccurate information about an individual. The goal is to prevent disclosure of identity in this setting.

Now assume that the query is broad, but still the database returns only a small number of records. What should we do in such a case? One easy solution is to refuse answering such queries. This is not effective, since the adversary can make several broader queries which contain the unanswered query range, and then take their intersection to determine which records belong to the unanswered range. Another easy solution is to append some fake data on the spot, so that the total number of records returned is at least k. This is not effective either since an adversary can make several similar queries and observe that only certain records are present in all of the returned results, thereby finding out that these are the only real data. However, this approach can be made effective if we can be consistent about the fake data. The idea is to insert a fixed set of fake data points all at once into the database such that the answer to any broad query either returns no records or at least k records.

We represent data records having multiple attributes as points in multidimensional space. This view is naturally suited for numeric data. We assume that queries are axis-parallel hyper-rectangles, which we will simply call d-dimensional boxes, that have certain minimum width in every dimension. The specific minimum width in each dimension can be different and needs to be chosen appropriately depending on the data. However, by appropriate scaling we can assure that the minimum width in every dimension is exactly one.

We would like the amount of fake data to be as small as possible. It is intuitive that in general the amount of fake data required is much smaller than the size of the database, since data is often densely concentrated in certain regions, and fake data is required only for

[*] Fachrichtung Informatik, Universität des Saarlandes, Im Stadtwald, Saarbrücken, 66123, Germany, alvarez@cs.uni-saarland.de

[†] Department of Mathematics and Computer Science, Saint Louis University, echambe5@slu.edu

[‡] Fachrichtung Informatik, Universität des Saarlandes, kozma@cs.uni-saarland.de

sparse regions. For example, if the multidimensional volume of the domain of the data is V, our broad query ranges have volume at least v, and there are n data points chosen uniformly at random from the domain, then the expected number of data points in a specific broad query range is nv/V. If this number is much larger than k, then with high probability a broad range is already k-anonymous, and needs no fake data. An easy calculation shows that the number of fake points required goes down exponentially with $nv/(kV)$.

1.1 Recasting to a geometric setting

The main geometric problem studied in this paper is the following: given a set P of n points in a d-dimensional box[1] $\mathbb{D} \subseteq [0, s]^d$ ($s \geq 1$), what is the smallest number of additional points that need to be added to P, so that every d-dimensional unit box contained in $\mathbb{D}$ is either empty or contains at least k points.

Notice that, if we want to hit all unit hypercubes, not just the non-empty ones, then this is a standard hitting set problem, with an obvious solution. The restriction to non-empty hypercubes is precisely what makes the problem difficult. This situation is similar to that of other hitting set problems, in which the ranges that we are interested in are defined implicitly. For example, when studying ε-nets, one is interested in hitting all ranges which have size at least εn. Implicitly defined hitting set problems also appear in combinatorial settings such as the feedback vertex set problem, where the goal is to pick the smallest set of vertices that hit all cycles of a graph.

No general technique is known to solve problems of this kind, and the methods for solving individual problems are varied. Our problem (defined formally in the next section) is motivated by the discussion in the previous section, and focuses on approximation algorithms for achieving k-anonymity.

1.2 Our contribution

Motivated by the above discussion, we define the following notions:

Definition 1 *A set P of n points contained in a box $\mathbb{D} \subseteq [0, s]^d$ ($s \geq 1$) is* k-anonymous*, for some given $k \geq 1$, if and only if every box of unit size contained in $\mathbb{D}$ is either empty or it contains at least k points of P.*

Note that any collection of points is trivially 1-anonymous. We therefore concern ourselves only with the case $k \geq 2$.

Definition 2 *Given a set P of n points as before, a* k-anonymizer *of P is a set $\mathcal{A} \subset \mathbb{D}$ of extra points such that $P \cup \mathcal{A}$ is k-anonymous.*

Our goal is to find a k-anonymizer of smallest cardinality. We call this an *optimal* k-anonymizer. The decision version of the problem is the following:

k-Anonymity:[2] Given a set P of n points as before and an integer l, is there a k-anonymizer of P of size at most l?

The results achieved in this paper are the following:

Theorem 1 k-ANONYMITY *is strongly NP-complete, even for $k = 2$.*

While we prove this for the two-dimensional case only, the result trivially implies the NP-completeness of the problem in any dimension $d \geq 2$. On the positive side, we give a polynomial-time approximation scheme (PTAS):

Theorem 2 *Let OPT denote the size of an optimal k-anonymizer for a set P of n points in $\mathbb{D} \subset \mathbb{R}^d$. Then, given $0 < \varepsilon \leq 1$, a k-anonymizer of P, of size at most $(1 + \varepsilon)OPT$ can be computed in $O((knd/\varepsilon)^{poly(k,(d/\varepsilon)^d)})$ time.*

Note that a fully polynomial time approximation algorithm (FPTAS) is not possible for strongly NP-complete problems, unless P=NP. Also, as the exponents in our approximation scheme are prohibitively large, we do not claim direct applicability of the algorithm, thus Theorem 2 should rather be taken as an existential result.

The rest of the paper is organized as follows: in Section 2 we introduce a dual setting, which is equivalent to the original problem but provides a better setting in which to prove our results. In Sections 3 and 4 we prove Theorems 1 and 2, respectively.

2 Dual setting

For convenience, we work in a “dual” setting based on our “primal” setting of input points/boxes, where we replace points by boxes and boxes by points. Each $p \in P$ gets mapped to the full dimensional unit box with its center at p, and every unit box $B \subset \mathbb{D}$ in the primal setting gets mapped to its center. This way, a set of n points P gets mapped to a set $\mathcal{Q}$ of n unit boxes. Observe that incidences between points and boxes are preserved by this transformation.

In all collections of points or boxes that we mention, we allow multiple copies of the same element. For simplicity we call these collections sets, even though technically they are multisets.

Given a set $\mathcal{Q}$ of n unit boxes, and a point $p \in \mathbb{D}$, we define the *depth* of p as the number of elements of $\mathcal{Q}$ that contain p. We now have the following dual definition of k-anonymity:

[1] In this paper, all boxes considered are axis-parallel.

[2] Our definition of k-ANONYMITY is slightly different from existing formulations in the literature, however, due to the strong similarity, we retained the term.

Definition 3 *Let $\mathcal{Q}$ be a set of n unit boxes of dimension d contained in a box $\mathbb{D} \subseteq [0, s]^d$ $(s \geq 1)$. We say that $\mathcal{Q}$ is a k-*anonymous arrangement of boxes *if and only if the depth of every point $p \in \mathbb{D}$ is either 0 or at least k.*

The equivalence between the two definitions follows from this simple observation: If p is a point and B is a unit box containing p, then the unit box centered at p contains the center of B.

Now the task is to find a set $\mathcal{A}$ of unit boxes (representing the extra points in the primal) of minimum cardinality such that $\mathcal{Q} \cup \mathcal{A}$ is k-anonymous. For completeness, we have the following formal definition of the decision version:

k-Anonymity (dual): Given a set $\mathcal{Q}$ of n unit boxes and an integer l, is there a k-anonymizer for $\mathcal{Q}$ with at most l boxes?

3 NP-completeness: proof of Theorem 1

First we show that k-Anonymity $\in$ NP. We consider an instance of the problem in the primal version (a set P of n points in a rectangle $\mathbb{D} \subset \mathbb{R}^2$ and a threshold l) and a candidate solution (a set $\mathcal{A}$ of t points in $\mathbb{D}$). We need to verify in time polynomial in $n + t$ that the solution is correct, i.e. $P \cup \mathcal{A}$ is k-anonymous and that $|\mathcal{A}| \leq l$. The latter can be checked with a simple counting, which takes $O(t)$ time. It remains to be shown that we can verify k-anonymity of a set of points in polynomial time.

Let us call a unit box in the primal setting a *test box*. There are an infinite number of locations in which a test box can be placed, but the following observation shows that it is sufficient to verify $O((n + t)^2)$ of them: If we move a test box continuously, the number of points inside does not change as long as the sides of the box do not cross any point. If we do not meet any points, we stop at the boundary of $\mathbb{D}$. It is therefore sufficient to look at test boxes in particular locations: one of the vertical sides of the test box touches a point or the boundary of $\mathbb{D}$ and one of the horizontal sides touches a point or the boundary of $\mathbb{D}$. By convention we do not count points on the top- or on the right side of the test box and we count points on the bottom- or on the left side as well as points inside the test box. Verifying that in all of these test boxes there are either at least k points or none, requires polynomial time.

Now we prove NP-hardness. This is done using a reduction from Planar3SAT, a known NP-complete decision problem [2]. In 3SAT, given a formula ϕ in 3-CNF, we ask whether there exists an assignment of truth values to the variables, such that ϕ evaluates to *true*. In Planar3SAT we restrict the question to planar formulae: those that can be represented as a planar graph in which vertices correspond to both variables and clauses of the formula and there is an edge between clause C and variable x if and only if C contains either x or $\neg x$.

Knuth and Ragunathan [4] observed that Planar3SAT remains NP-complete if we restrict it to formulae having the following *rectilinear embedding*: variables are placed on a line, clauses are placed on the two sides of the line and the three legs of each clause are properly nested (Figure 1(a)).

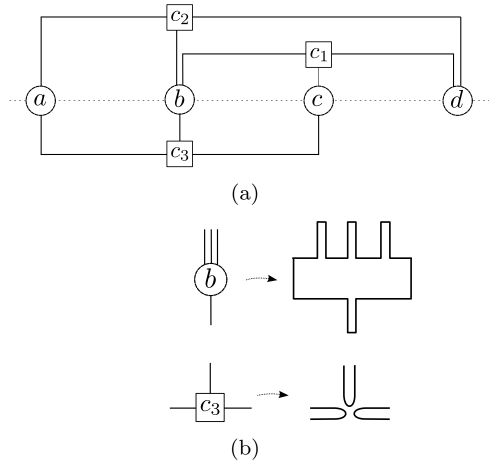

Figure 1: (a) Rectilinear embedding of the formula $(b \vee \neg c \vee \neg d) \wedge (a \vee \neg b \vee d) \wedge (\neg a \vee b \vee c)$.
(b) A variable with three connections on top and one at the bottom and its corresponding gadget (above). Clause and its gadget (below).

Given an instance of Planar3SAT with an embedding as described before, we transform it into a two dimensional instance of the (dual) k-Anonymity decision problem.

The unit squares of the set $\mathcal{Q}$ are placed so as to align with an orthogonal grid with cells of size $\frac{1}{5} \times \frac{1}{5}$. The placement of $\mathcal{Q}$ will create the following types of regions in $\mathbb{D}$: (a) *empty* regions, consisting of points with depth 0, (b) *uncovered* regions consisting of points with positive depth less than k, which need to be "fixed" by the k-anonymizer and (c) *safe* regions consisting of points with depth at least k. Our construction will assure that all *uncovered* regions have a depth of exactly $k - 1$, therefore we need not add multiple copies of the same square in the solution.

We can create an *uncovered* square of size $\frac{1}{5} \times \frac{1}{5}$ in the following way: we put $k - 1$ squares in the same place, k squares shifted to the right by $\frac{1}{5}$ and k squares shifted upwards by $\frac{1}{5}$. We call the resulting uncovered square a *patch* and it is the main element in our reduction.

Our construction is such that *patches* are surrounded by large *safe* regions. Consider a box $B \in \mathcal{A}$ that covers one or more patches created by the input set $\mathcal{Q}$. The parts of B that do not cover a patch fall entirely within the safe region surrounding the patches. In this way we ensure that the boxes in the k-anonymizer $\mathcal{A}$ do not create new uncovered regions.

Our main gadget is a sequence of patches which we call a *wire* (Figure 2). Note that the wire can be extended infinitely at both ends. The wire has two important properties: (1) a unit square can cover any two neighboring patches, such that the leftover part falls entirely in the safe region, and (2) no unit square can cover more than two patches of the wire. In Figure 2, uncovered patches are colored black, the surrounding safe region is gray. We also show the corresponding points in the primal diagram, together with their multiplicities.

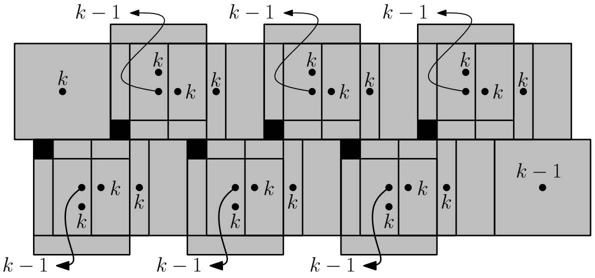

Figure 2: Wire gadget in dual version and primal points with their multiplicities.

We can also design a *bend* gadget, which will introduce 90 degrees turns in a wire, shown in Figure 3 in the Appendix. Both of the previously mentioned properties of a wire are also preserved by the bend gadget, which we illustrate with dotted unit squares that cover neighboring patches..

Now we can create loops and we will represent each variable by a loop that contains an even number of patches. Let the patches in a loop be numbered from 1 to $2m$. It can be observed that the optimal k-anonymizer of this loop has m boxes, each covering two neighboring patches. The boxes cover either the patches $(1, 2), (3, 4), \ldots, (2m - 1, 2m)$, or the patches $(2, 3), (3, 4), \ldots, (2m, 1)$. The choice between the two optimal solutions encodes the truth value of a variable.

We transmit the value of a variable with a *tentacle* extending from the main loop. A tentacle consists of two parallel wires with a sufficient distance between them to avoid interference. A clause gadget is the meeting point of three such tentacles. The variable and the clause gadget are schematically presented in Figure 1(b). The line here indicates a wire, without showing the actual patches.

We will show the clause gadget in more detail in the full version of this paper. However, as mentioned before, it is the meeting point of three variable tentacles. Besides the patches that make up the variable tentacles, the clause contains an extra patch in the middle. This patch is placed in such a way that it is reachable by a square of the optimal covering of either of the three variables, but only if the variable is in one particular state. By convention, we consider this the *true* state. This means that if all three variables are *false*, we need an extra square to cover the patch in the middle. This penalty is the key ingredient of our reduction.

We have not yet discussed negated variables. If a variable appears negated in a clause, we need to lengthen the corresponding tentacle by one patch on each side, so that the two patches nearest to the clause center are now covered by a single square in the *false* state. We can achieve such a shift by replacing a small piece of a wire by a *condensing* gadget. This gadget increases the number of patches by one, while keeping the endpoints in place and maintaining properties (1) and (2) of a wire. We omit the details of this gadget, as it is a straightforward construction.

Our reduction is almost complete; what is left is the computation of the parameter l in the k-Anonymity instance. Let the total number of patches in wires (excluding the extra patches in the middle of clauses) be $2m$. Then we set $l = m$ and conclude that there exists a k-anonymizer of size at most l if and only if the original Planar3SAT instance has a satisfying assignment.

For completeness, we need to prove that our construction is of polynomial size (in terms of the number of clauses and variables of the Planar3SAT instance). First we observe that the resulting construction can be bounded by a box of polynomial size: the height of the box depends on the maximum level of nesting in the embedding of the formula, but each additional level results in an increase of constant size. The number of levels is bounded by the number of clauses in the formula. The width of the rectangle increases with the addition of each new variable or clause, but again, only by a constant additive term. Since our construction consists of points with multiplicity at most k aligned with a grid of size $\frac{1}{5} \times \frac{1}{5}$, the total number of points needed is less than $25k$ times the size of the bounding box, which is clearly polynomial. This concludes the proof of Theorem 1.

4 PTAS: proof of Theorem 2

As before, all this section will take place in the dual setting, where the input set of points P is represented by an arrangement of boxes $\mathcal{Q}$. The main result of this section is the proof of correctness of the following randomized algorithm that computes a k-anonymizer of size at most $(1 + \varepsilon)OPT$, where OPT denotes the size of an optimal k-anonymizer of $\mathcal{Q}$. This algorithm is based on a technique developed by Hochbaum and Maass [1]. Additionally, at the end of the section we will discuss how to derandomize our algorithm..

```
Algorithm 1
    Input: A set $\mathcal{Q}$ of $n$ unit boxes in $\mathbb{D} \subseteq [0, s]^d$ ($s \geq 1$).
    Output: An anonymizer of size $(1 + \varepsilon)OPT$ for $\mathcal{Q}$.
1.  Given $0 < \varepsilon \leq 1$, choose $L = (2d/\varepsilon)$ and a random integer
    $r \in [0, L - 1]$.
2.  Impose a grid $\mathbb{G}$ over domain $\mathbb{D}$ of cell size $L$, and with offset $r$
    from the origin in every dimension.
3.  Find an optimal $k$-anonymizer inside every non-empty cell of $\mathbb{G}$.
4.  Output the union of the solutions of the cells of $\mathbb{G}$.
```

In Step 2 of the algorithm, offset r from the origin means that the coordinates of every grid point inside domain $\mathbb{D}$ are of the form $(r+cL)$, where $c \geq 0$ is integer. The grid points on the boundary of $\mathbb{D}$ are automatically defined. Note that *non-empty* refers to the dual view: we consider a cell empty if all its points have depth 0 or at least k.

As the reader can note, the only step in the algorithm that is non-trivial is Step 3, in which we have to compute an exact solution of a subproblem contained in a smaller domain. In order to make the presentation simpler, we first focus on the case $d = 2$. We then look at the general case, and then consider derandomization of Algorithm 1.

Given two axis-parallel squares, we say that they are *aligned* if and only if they intersect only at their boundaries, i.e. their intersection is non-empty, but they have disjoint interiors. Now, given a set of unit squares $\mathcal{Q} = \{Q_1, \ldots, Q_n\} \subset \mathbb{D} \subseteq [0, s]^2$, we can define a grid $\mathbb{G}_{\mathcal{Q}}$ over $\mathbb{D}$ as follows:

Let $Q \in \mathcal{Q}$ and define $\mathbb{G}^Q$ to be the unit grid over $\mathbb{D}$ having Q as a cell. Denote by $E(\mathbb{G}^Q)$ the set of grid lines of $\mathbb{G}^Q$. The set of grid lines of $\mathbb{G}_{\mathcal{Q}}$ is $\bigcup_{i=1}^{n} E(\mathbb{G}^{Q_i})$, and its vertex set is the set of intersection points among its grid lines. We now have the following lemma:

Lemma 3 *Let $\mathcal{Q} \subset \mathbb{D}$ and $\mathbb{G}_{\mathcal{Q}}$ be as defined before. Then there exists an optimal k-anonymizer $\mathcal{A}$ of $\mathcal{Q}$ such that each of its elements has its vertices at grid points of $\mathbb{G}_{\mathcal{Q}}$.*

Proof. We show that there is an optimal k-anonymizer that is aligned in the horizontal direction. By a similar argument it can be shown that there is an optimal k-anonymizer that is aligned in both the horizontal and vertical directions.

Let us proceed by contradiction. Let $\mathcal{A}$ be the optimal k-anonymizer with the least number of elements whose vertical sides are not aligned with the vertical grid lines of $\mathbb{G}_{\mathcal{Q}}$. Denote by U this set of "unaligned" elements of $\mathcal{A}$. If $U = \emptyset$ then we are done, so we will assume that $U \neq \emptyset$. If we manage to move the elements of $\mathcal{A}$ around, such that the cardinality of U decreases, we are done.

We will move the boxes in U to the right simultaneously and at the same speed until we are about to de-anonymize some region and are forced to stop. At that point, some element of U gets aligned (horizontally) with some element $B \in \mathcal{Q} \cup (\mathcal{A} \setminus U)$, and thus automatically with a vertical grid line of $\mathbb{G}_{\mathcal{Q}}$, since B was already aligned. Observe that during this process we do not de-anonymize any region of $\mathbb{D}$ but we "align" one element of U. This is a contradiction since we assumed that U was of minimum cardinality.

Once we have an optimal solution that is horizontally aligned to $\mathbb{G}_{\mathcal{Q}}$ we can repeat the argument with a solution whose boxes are horizontally aligned and a minimum number of them are vertically unaligned. □

We can now perform Step 3 of Algorithm 1 in polynomial time:

Lemma 4 *Let $\mathcal{Q} = \{Q_1, \ldots, Q_n\}$ be a set of unit squares contained in a square of side-length L. Then a k-anonymizer of minimum cardinality can be found in time $O\left((knL)^{poly\left(k, L^2\right)}\right)$.*

The proof is based on Lemma 3. Since there exists an optimal k-anonymizer in which every element is aligned with the grid, we exhaustively search all candidate sets in increasing order of size, until we find a solution. We defer the details to the full version of this paper.

Note that in practice L and k might be small, and independent of n, giving a running time of the sort $O(n^c)$, for some constant c, which is thus polynomial in n. We now prove a result that implies Theorem 2 for the case $d = 2$, by setting $L = 4/\varepsilon$.

Theorem 5 *Let $\mathcal{Q} \subset \mathbb{D}$ be a set of n unit squares defined as before. Then Algorithm 1 computes a k-anonymizer of $\mathcal{Q}$ of size at most $(1 + \varepsilon)OPT$ in time $O\left((knL)^{poly\left(k, L^2\right)}\right)$.*

Proof. To achieve the desired running time, we run the algorithm of Lemma 4 in each non-empty cell of the grid $\mathbb{G}$ imposed over domain $\mathbb{D}$ in Step 2 of Algorithm 1. Since there are at most n non-empty cells, the overall running time is $O((knL)^{poly(k, L^2)})$. Observe that the cell-size is at most $L = \frac{4}{\varepsilon}$, which is independent of n.

As for the quality of approximation, let $0 < \varepsilon \leq 1$ be a given parameter. Let OPT denote the size of an optimal solution $\mathcal{A}$. By the previous discussion, we know that each element of $\mathcal{A}$, a unit square, can intersect (1) one vertical line and no horizontal line, or (2) one horizontal line and no vertical line, or (3) one vertical and one horizontal line, i.e. it can contain exactly one grid point of $\mathbb{G}$, or (4) lie entirely in a cell of $\mathbb{G}$. Now consider some $q \in \mathcal{A}$. If q is of type (1) or (2), note that q then intersects exactly two cells C, C' of $\mathbb{G}$. In this case, we will create another copy q' of q and move q to C and q' to C' taking care that $\mathcal{A} \cup \{q'\}$ remains a k-anonymizer, although not of minimum cardinality anymore. If q is of type (3), then we will create three more copies of it and distribute the four of them among the four cells of $\mathbb{G}$ that q intersects. By performing these operations on every element of $\mathcal{A}$ of type (1), (2), or (3) we create a new k-anonymizer $\mathcal{A}'$ with the property of having each element inside some cell of $\mathbb{G}$.

Let us denote by OPT' the size of the solution obtained by Algorithm 1. Let C be a cell of $\mathbb{G}$ and observe that the local solution given by $\mathcal{A}'$ on C must be at least as large as the local solution provided by Algorithm 1, since the latter is optimal for C. Therefore we obtain that $OPT' \leq |\mathcal{A}'|$. We can think of $\mathcal{A}'$ as a version of $\mathcal{A}$ with a penalty. Note that by the random offset r of $\mathbb{G}$, an element Q of $\mathcal{A}$ can intersect a vertical or horizontal line with probability $\frac{1}{L}$, since r takes values from 0 to $L-1$, and Q is a *unit* square, so Q gets penalized only in one out of the L unit intervals of $[0, L]$. Note as well that Q intersects a vertical and a horizontal line independently, so the expected penalty of Q is $1 \cdot p_1 + 1 \cdot p_2 + 3 \cdot p_3 + 0 \cdot p_4$, where p_i is the probability that Q is of type (i). We obtain (using the fact that $L = \frac{4}{\varepsilon}$):

$$\begin{aligned}\mathbb{E}[\text{Penalty of } Q] &= \frac{1}{L}\left(1-\frac{1}{L}\right)+\frac{1}{L}\left(1-\frac{1}{L}\right)+\frac{3}{L^2}\\ &= \frac{2}{L}+\frac{1}{L^2}\leq\frac{3}{L}.\end{aligned}$$

Therefore, $\mathbb{E}[OPT'] \leq \mathbb{E}[|\mathcal{A}'|] \leq |\mathcal{A}| + |\mathcal{A}| \cdot \frac{3}{L} = \left(1+\frac{3}{L}\right) OPT < (1+\varepsilon)OPT$.

□

Note that all arguments carry over to higher dimensions. We summarize the result in the following theorem, which implies Theorem 2 for general dimension d by setting $L = 2d/\varepsilon$. We leave the details of the proof for the full version of this paper.

Theorem 6 *Let $\mathcal{Q} \subset \mathbb{D}$ be a set of n unit boxes in d dimensions, and let $L = \frac{2d}{\varepsilon}$. Then Algorithm 1 computes a k-anonymizer of $\mathcal{Q}$ of size at most $(1+\varepsilon)OPT$ in time $O\left((kdnL)^{poly\left(k,L^d\right)}\right)$.*

Finally, observe that the random offset r is an integer in the interval $[0, L-1]$. Since the expected size of the computed k-anonymizer is $(1+\varepsilon)OPT$, we can try each possible value of r, and pick the smallest k-anonymizer. This derandomizes Algorithm 1 adding a factor of L to the running time.

5 Conclusions

In this paper, we presented a new notion of k-anonymity that uses fake data to achieve anonymity assuming that queries are *broad*. We studied the complexity of the associated optimization problem in a geometric framework, which allowed us to leverage techniques available in computational geometry. We proved strong NP-completeness and gave a PTAS in fixed dimensions and for constant k. Note that this is mostly of theoretical interest: the exponent in the running time is very large, even in two dimensions and for small k. It is still not clear whether the exact dependence on the number of data points can be improved. One can easily imagine a situation where the number of points is arbitrarily large but the optimal solution remains the same for a much smaller subset. It may be possible to sample a subset of the input points, from which with high probability a good approximation may be obtained by using our algorithm on the sample.

Acknowledgements

We thank Sasanka Roy and Saurabh Ray for suggesting the problem and for fruitful discussions. Victor Alvarez was partially supported by CONACYT-DAAD of México. Erin Chambers is partially supported by NSF grant CCF 1054779.

References

[1] D. Hochbaum, W. Maass, *Approximation schemes for covering and packing problems in image processing and VLSI*, J. ACM, **31:1**:130–136, 1985.

[2] D. Lichtenstein, *Planar formulae and their uses*, SIAM J. Comput., **11:2**:329–343, 1982.

[3] L. Sweeney, *k-Anonymity: A Model for Protecting Privacy*, Intl J. of Uncertainty, Fuzziness and Knowledge-Based Systems, **10:5**:557–570, 2002.

[4] D.E. Knuth, A. Raghunathan, *The problem of compatible representatives*, SIAM J. Discret. Math., **5:3**:422–427, 1992.

[5] A. Meyerson, Ryan Williams, *On the Complexity of Optimal K-Anonymity*, Proc. of the Twenty-third ACM SIGACT-SIGMOD-SIGART, 223–228, 2004.

[6] K.LeFevre, D. J. DeWitt, R. Ramakrishnan, *Mondrian Multidimensional K-Anonymity*, Proc. of the 22nd Intl Conf. on Data Engineering, ICDE 2006, 3-8 April 2006, Atlanta, GA, USA

[7] C. Dwork, F. McSherry, K. Nissim, and A. Smith, *Calibrating noise to sensitivity in private data analysis*, Proc. of the 3rd Theory of Cryptography Conf., 265–284, 2006.

Appendix A. Figures

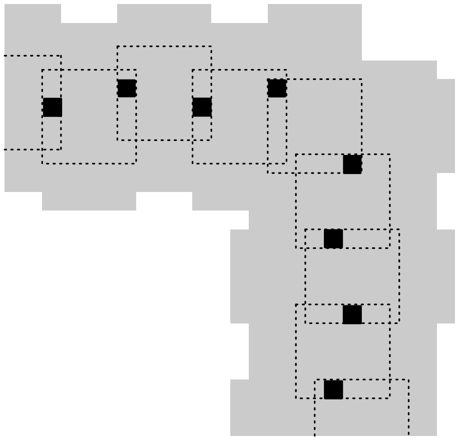

Figure 3: Bend gadget (dual) and unit squares that cover neighboring patches.

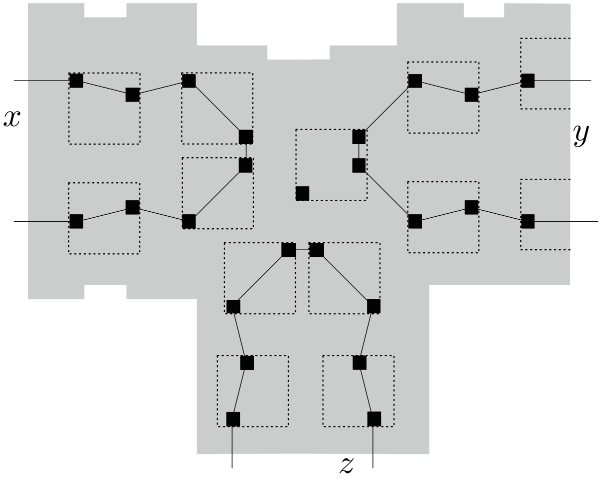

Figure 4: Clause gadget (dual) corresponding to clause $(x \vee y \vee z)$ and k-anonymizer encoding the assignment $x = false, y = true, z = false$.

Appendix B. Proofs

Proof of Lemma 4 By Lemma 3 we know that there exists a k-anonymizer of minimum cardinality whose elements have their vertices at grid points of $\mathbb{G}_{\mathcal{Q}}$, so it is enough for us to look for candidate elements only at those locations. However, we do not know how big the smallest k-anonymizer will be, but we know that it can not have size larger than kL^2, since a trivial solution would be to impose a unit grid over the domain and use k times each of the L^2 cells of this grid to be sure that no region of $\mathcal{Q}$ stays non-anonymized.

With the previous conditions on the location and the size of the k-anonymizer that we are looking for, we just have to search using brute force over all subsets of grid points of $\mathbb{G}_{\mathcal{Q}}$ up to size kL^2, taking into consideration that each element of this subset could appear at most k times in an optimal solution, and keep the solution of minimum size. By convention, when choosing a grid point, we will choose the unit cell of $\mathbb{G}_{Q}$ having this grid point as the north-west corner, observe that this technicality does not affect the quality of our solution.

The number of grid points in $\mathbb{G}_{\mathcal{Q}}$ is can be easily checked to be $O\left((nL)^2\right)$. Now, since each grid point can appear up to k times, we can create k copies of each grid point, thus the number of candidate subsets of grid points of $\mathbb{G}_{\mathcal{Q}}$ we have to check is at most:

$$\sum_{i=0}^{kL^2} \binom{O\left(k(nL)^2\right)}{i} = O\left(\left(k(nL)^2\right)^{kL^2+1}\right)$$

Note however that each time we try a subset of size i of grid points we have to check if it is a candidate for a solution. This can be done in polynomial time in i and n since k-Anonymity belongs to NP, which we proved in Section 3. The overall running time is thus of the form $O\left((knL)^{poly\left(k,L^2\right)}\right)$.

Proof of Theorem 6 We proceed in an analogous way to the proofs of Lemma 4 and Theorem 5, but in d-dimensions, for $d \geq 3$, we define $\mathbb{G}_{\mathcal{Q}}$ not as a grid, but as an arrangement of hyperplanes.

Let $Q \in \mathcal{Q} \subset \mathbb{D}$ be a d-dimensional unit box, still axis-parallel. Box Q defines the following arrangement of hyperplanes: let H_i be the hyperplane supporting one of the sides of Q in direction $1 \leq i \leq d$. By suitable rotation of the space we can assume w.l.o.g. that H_i is a "vertical" hyperplane. Make parallel copies of H_i to the left and right of H_i, each one at unit distance from the previous one, until we get out of the domain $\mathbb{D}$ that Q is contained in. At this point we ignore the hyperplanes outside $\mathbb{D}$ but we add the two hyperplanes supporting the two sides of $\mathbb{D}$ that are parallel to H_i. If $\mathbb{D} \subseteq [0, s]^d$ $(s \geq 1)$ then there are $O(s)$ hyperplanes spawned by $\mathcal{Q}$ in direction $1 \leq i \leq d$, and thus $O(d \cdot s)$

hyperplanes spawned by Q in all d directions. This last set of hyperplanes is the arrangement of hyperplanes defined by Q. Hence, in d-dimensions, $\mathbb{G}_{\mathcal{Q}}$ is the union of the arrangements of hyperplanes spawned by *all* elements of $\mathcal{Q}$, along with all their lower-dimensional intersections.

The reader will be able to verify now that Lemma 3 also holds with this new definition of $\mathbb{G}_{\mathcal{Q}}$, since the elements of a k-anonymizer of $\mathcal{Q}$ are aligned one dimension at a time, until they all sit at vertices of $\mathbb{G}_{\mathcal{Q}}$. As for Lemma 4, it is known that the complexity of an arrangement of m hyperplanes in d-dimensions is $\Theta(m^d)$. It is easy to check that $m = O(d \cdot n \cdot L)$ if $\mathcal{Q}$ is contained in a box of side-lenght $L \geq 1$. Thus the d-dimensional algorithm of Lemma 4 would run in time $O\left((kdnL)^{poly\left(k,L^d\right)}\right)$ since we only have to check sub-sets of vertices of $\mathbb{G}_{\mathcal{Q}}$.

Algorithm 1 imposes a grid $\mathbb{G}$ of cell size L over the domain $\mathbb{D}$. The d dimensional version of Lemma 4 can be run on each non-empty cell of $\mathbb{G}$, thus giving an overall running time of $O\left((kdnL)^{poly\left(k,L^d\right)}\right)$ since there are at most n non-empty cells of $\mathbb{G}$.

As for the quality of the approximation, we now observe that an element of an optimal solution $\mathcal{A}$ has $\binom{d}{1}$ possibilities of being shared by two cells, $\binom{d}{2}$ of being shared by four cells, and in general $\binom{d}{i}$ possibilities of being shared by 2^i cells, with $0 \leq i \leq d$. Hence, the penalty in each case is $2^i - 1$. Thus the expected penalty of an element $Q \in \mathcal{A}$ is:

$$
\begin{aligned}
&\mathbb{E}[\text{Penalty of } Q] = \\
&\quad = \sum_{i=0}^{d}(2^i - 1)\binom{d}{i}\mathbb{P}[Q \text{ shared by exactly } 2^i \text{ cells}] \\
&\quad = \sum_{i=0}^{d}(2^i - 1)\binom{d}{i}\left(\frac{1}{L}\right)^i\left(1 - \frac{1}{L}\right)^{d-i} = \left(1 + \frac{1}{L}\right)^d - 1 \\
&\quad = \sum_{i=1}^{d}\binom{d}{i}\left(\frac{1}{L}\right)^i \leq \sum_{i=1}^{d}\left(\frac{d}{L}\right)^i \leq \frac{d}{L}\sum_{i=0}^{\infty}\left(\frac{d}{L}\right)^i \\
&\quad = \frac{d}{L}\left(\frac{1}{1 - \frac{d}{L}}\right) = \frac{d}{L - d} = \frac{\varepsilon}{2 - \varepsilon} \leq \varepsilon.
\end{aligned}
$$

The last two inequalities follow from the fact that $L = \frac{2d}{\varepsilon} > d$, and $0 < \varepsilon \leq 1$.

Again, we obtain:

$$\mathbb{E}[OPT'] \leq |\mathcal{A}| + |\mathcal{A}| \cdot \varepsilon = (1 + \varepsilon)OPT.$$