Finding Quasi-identifiers Against k-anonymity

Chih-chen Yuan

National Taiwan Normal University, Taiwan

Abstract. 我是資工系大四的袁至誠，我的學號是 40371115H。This article briefly summarizes some current results of my ongoing work in privacy research with my teacher 王柏堯. I thought I'd use this opportunity to fulfill the essay requirement of 全球化與社會關懷課程 while also reviewing my own work.

1 Introduction

The field of data science has made tremendous strides in the past decade, thanks to the maturation of hardware and the widespread adoption of data gathering devices, such as environment sensors and social networks. The sheer amount of data from various sources combined could potentially reveal otherwise unauthorized private information about people. For example, the data used for Netflix Prize, an open competition on collaborative filtering algorithm, was found by Narayanan and Shmatikov [7] to suffer from serious privacy vulnerabilities, costing Netflix a lawsuit thereafter.

Sweeney [8] proposed one of the first privacy protection model of data releasing. With the advent of big data and machine learning, the demand of such techniques increases with each passing years, as more data providers seek ways to employ new technologies while upholding the promises of protecting user data.

In [8], Sweeney demonstrated that, at as early as 1997, the voter registration data of Cambridge, Massachusetts sold by the city government and the health insurance data of state employees sold by the Group Insurance Commission (GIC) can be linked together to uniquely re-identify some personal medical information with only ZIP code, birth date, and gender. For example, the governor of Massachusetts then, William Weld, is the only man with that particular birth date in that ZIP code.

To protect released data from this kind of de-identification by attribute linking, Sweeney formalized the notion of k-anonymity upon the idea of quasi-identifier (QID) which is some set of potentially sensitive attributes capable of re-identification, introduced by Dalenius [4] in 1986. For some integer k, to say some QID is k-anonymous is to say that there are at least k entries with identical values, for all combinations of values present.

Since its conception, to date, potential attacks were discovered and more sophisticated paradigms, such as l-diversity, t-closeness, or more recently, differential or transparent approaches were developed. However, in this work, we shall focus on the fundemental idea of k-anonymity and consider some problems defined on it.

The most natural problem associated with k-anonymity is the problem of, given a database, transforming it so that it conforms to some level of k-anonymity. Meyerson and Williams [6] proved that finding such solution is NP-hard for 3-anonymity with unrestricted attribute domains. Later, Aggarwal et al. [1] and Dondi et al. [3], respectively, proved the NP-hardness of 3-anonymity with attribute domains restricted to 3 and 2. For the case of 2-anonymity, Blocki and Williams [2] proved that it's in P. The complexity results of general case call for approximation algorithms and several have also been developed.

In this work, I consider the more pragmatic problem of deciding if some given database conforms to k-anonymity and that, if it does not, what are the violating QIDs.

2 Preliminaries

I shall first give the formal definition of the concepts I'll be working on.

Definition 1. *A database $\mathcal{D}$ is a finite set of m-tuples associated with an m-tuple of column names $\mathcal{C}$ and an m-tuple of finite sets of column domains $\mathcal{F}$, such that $\mathcal{D} \subseteq \bigtimes_{F \in \mathcal{F}} F$ and $|\mathcal{D}| \leq |F|$ for all $F \in \mathcal{F}$.*

Definition 2 (k-anonymity). *A database $\mathcal{D}$ is k-anonymous iff, for all $d \in \mathcal{D}$ and for all $c \subseteq \mathcal{C}$, there exist a subset $S \subseteq \Pi_c \mathcal{D}$ of size at least k such that $s = \Pi_c d$ for all $s \in S$.*

Definition 3. *A quasi-identifier (QID) is a finite set of columns $Q \subseteq \mathcal{C}$ such that it is a witness to violation of k-anonymity; that is, there is some $d \in \mathcal{D}$ such that there is no $S \subseteq \Pi_Q \mathcal{D}$ with $s = \Pi_Q d$ for all $s \in S$ of size at least k.*

Now, I am ready to define the problems at hand.

3 *k*-anonymity-test

Consider the problem of *k-anonymity-test*, which decides if some given database is k-anonymous.

Problem 1 (k-anonymity-test). Given an instance of a database $\mathcal{D}$, is $\Pi_c \mathcal{D}$ k-anonymous for all $c \subseteq \mathcal{C}$.

For the following, I shally employ Sweeney's Lemma from [8], which effectively states that removing columns can only make it k-anonymous and that, vice versa, adding columns can only make it not k-anonymous.

Lemma 1. *Given an instance of a database $\mathcal{D}$, if $\Pi_A \mathcal{D}$ is k-anonymous, then $\Pi_B \mathcal{D}$ is k-anonymous for all $B \subseteq A$.*

Proposition 1. *k-anonymity-test is in P.*

Proof. Given $\mathcal{D}$, by Lemma 1, it suffices to solve k-anonymity-test by checking whether $\Pi_{\mathcal{C}} \mathcal{D}$ is k-anonymous, which is polynomial in the size of $\mathcal{D}$.

Therefore, to check if k-anonymity holds for all subsets of columns for a given database, we only need to check if k-anonymity holds for the case of the whole sets of columns.

4 minimum-QID

If some database is found to not be k-anonymous, the natural question to ask is: what are the QIDs that violates k-anonymity? By Lemma 1, since all supersets of violating QIDs are themselves violating QIDs, we shall consider the problem of finding the minimum QIDs.

Definition 4. *For a QID Q that violates k-anonymity, it is minimum if Q' does not violate k-anonymity for all $|Q'| < |Q|$.*

Note that minimum QID is unique. Now we can look at the function problem presented by this definitions.

Problem 2 (minimum-QID). Given an instance of a database $\mathcal{D}$, find the minimum QID.

Proposition 2. *minimum-QID is NP-hard.*

4.1 Proof

As function problems can be easily transformed to corresponding decision problems. To prove the NP-hardness, I consider the decision version of the problem, k-QID: if there is some QID of size k, and reduce it to the minimum cover problem, a NP-complete problem, which is [SP5] in Garey and Johnson [5]. It asks: given a collection C of subsets of a finite set S and a positive integer $K \leq |C|$, does C contain a cover for S of size K or less, i.e., a subset $C' \subseteq C$ with $|C'| \leq K$ such that every element of S belongs to at least one member of C'?

The construction goes as follows. Given an instance of minimum cover, construct a database $\mathcal{D}$ with $\mathcal{C}$ as columns and $S \cup \{\perp\}$ as rows. For the content of the database, let

$$\mathcal{D}[i][j] = \begin{cases} i, & \text{if } S[i] \in \mathcal{C}[j] \\ \perp, & \text{otherwise} \end{cases}$$

where $\mathcal{D}[i][j]$ represents the jth element in the ith tuple in $\mathcal{D}$ and $\perp$ is a fresh symbol not in S.

Now for the correctness that the given instance satisfies minimum cover iff the constructed instance satisfies k-QID.

Suppose there is such subset C' of C such that every element in S is in at least one set of C'. Project the columns corresponding to C' in the database. Since every element is in at least one set of C', every row except the $\perp$-row has at least one element unique to that row, so there are no duplicate rows in $\Pi_{C'}\mathcal{D}$; thus, the number of rows in the projected database $\Pi_{C'}\mathcal{D}$ is equal to the original database $\mathcal{D}$.

Suppose there is some subset of columns N such that the number of rows in the projected database $\Pi_N \mathcal{D}$ is equal to the original database $\mathcal{D}$. Per the assumption, it immediately follows that there are no duplicate rows in $\Pi_N \mathcal{D}$; furthermore, per the construction rule of the database, every row must have at least one non-$\perp$ element in some column in N. Thus, choose those subsets C' corresponding to N from C, and every element of S must be in at least one set in C'.

4.2 Example

Let $k = 2$, $S = \{a, b, c, d, e\}$, and $C = \{c_1, c_2, c_3, c_4\}$ with $c_1 = \{a, b, c\}, c_2 = \{b, d\}, c_3 = \{c, d\}, c_4 = \{d, e\}\}$. There is a solution $C' = \{c_1, c_4\} = \{\{1, 2, 3\},\ \{4, 5\}\}$.

The constructed database for this instance is:

$$
\begin{array}{rccccl}
 & c_1 & c_2 & c_3 & c_4 & \\
\mathcal{D} = \{ & (1, & \perp, & \perp, & \perp), & a \\
 & (2, & 2, & \perp, & \perp), & b \\
 & (3, & \perp, & 3, & \perp), & c \\
 & (\perp, & 4, & 4, & 4), & d \\
 & (\perp, & \perp, & \perp, & 5), & e \\
 & (\perp, & \perp, & \perp, & \perp)\} & \perp
\end{array}
$$

If we project the solution, i.e., $\Pi_{c_1, c_4} \mathcal{D}$, we get:

$$
\begin{array}{rccl}
 & c_1 & c_4 & \\
\mathcal{D} = \{ & (1, & \perp), & a \\
 & (2, & \perp), & b \\
 & (3, & \perp), & c \\
 & (\perp, & 4), & d \\
 & (\perp, & 5), & e \\
 & (\perp, & \perp)\} & \perp
\end{array}
$$

where no two rows are identical, meaning every element is in at least one of c_1, c_4.

If we project something incorrect, e.g., $\Pi_{c_1,c_2,c_3}\mathcal{D}$, we get:

$$\begin{array}{rcccl} & c_1 & c_2 & c_3 & \\ \mathcal{D} = \{(1, & & \perp, & \perp), & a \\ & (2, & 2, & \perp), & b \\ & (3, & \perp, & 3,), & c \\ & (\perp, & 4, & 4,), & d \\ & (\perp, & \perp, & \perp), & e \\ & (\perp, & \perp, & \perp)\} & \perp \end{array}$$

where the e-row and $\perp$-row are identical, meaning e is not in any of c_1, c_2, c_3.

5 minimal-QID

As minimum-QID is NP-hard, I turned to try to come up with a randomized algorithm and, in turn, found a solution to a somewhat slightly different problem, where the result is only locally optimal, the minimal-QID problem.

Definition 5. *For a QID Q that violates k-anonymity, it is minimal if Q' does not violate k-anonymity for all $Q' \subseteq Q$.*

Note that a minimal QID is not unique as it may be only locally maximal.

Problem 3 (minimal-QID). Given an instance of a database $\mathcal{D}$, find the minimal QID.

Proposition 3. *minimal-QID is in P.*

I shall prove it by simply giving the polynomial time algorithm for finding a minimal QID. This algorithm essentially avoid the factorial blow-up of combinations by choosing only the immediately successing level and descending in a greedy manner whenever possible.

5.1 Algorithm

Data: A database $\mathcal{D}$.
Result: A set of columns that completely identifies the database.

```
goal = |Π_C D|
fields = C
found = True
while found do
    found = False
    for next ∈ combinations(fields, |fields| − 1) do
        if |Π_next D| = goal then
            found = True
            fields = next
            break
return fields
```

The running time is $\mathcal{O}(\Sigma_{i=1}^{|\mathcal{D}|}\binom{i}{i-1}) = \mathcal{O}(|\mathcal{D}|^2)$.

5.2 Example

Consider this database.

a	b	c	d	e
x	1	x	6	x
x	x	2	7	x
x	3	x	8	x
x	x	4	x	9
5	x	x	x	0

The minimum QID is (d,e) and the algorithm finds the minimal QID (a,b,c) following these steps.

a	b	c	d	e	yes
a	b	c	d		yes
a	b	c			yes
a	b				no
	b	c			no
a		c			no

6 Conclusion

I should be able to fashion the algorithm for finding minimal QID into some sort of randomized algorithm with proper bounds for finding minimum QID. This is going to be the next step for my research.

References

1. Aggarwal, G., Feder, T., Kenthapadi, K., Motwani, R., Panigrahy, R., Thomas, D., Zhu, A.: Approximation algorithms for k-anonymity. In: Proceedings of the International Conference on Database Theory (2005)
2. Blocki, J., Williams, R.: Resolving the complexity of some data privacy problems. In: Proceedings of the 37th International Colloquium Conference on Automata, Languages and Programming: Part II. pp. 393–404. ICALP'10 (2010)
3. Bonizzoni, P., Della Vedova, G., Dondi, R.: The k-anonymity problem is hard. In: Kutyłowski, M., Charatonik, W., Gębala, M. (eds.) Fundamentals of Computation Theory. pp. 26–37. Springer Berlin Heidelberg, Berlin, Heidelberg (2009)
4. Dalenius, T.: Finding a needle in a haystack or identifying anonymous census records. Journal of Official Statistics **2**(3), 329–336 (1986)
5. Garey, M.R., Johnson, D.S.: Computers and Intractability; A Guide to the Theory of NP-Completeness. W. H. Freeman & Co., New York, NY, USA (1990)
6. Meyerson, A., Williams, R.: On the complexity of optimal k-anonymity. In: Proceedings of the Twenty-third ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems. pp. 223–228. PODS '04 (2004)
7. Narayanan, A., Shmatikov, V.: Robust de-anonymization of large sparse datasets. In: Proceedings of the 2008 IEEE Symposium on Security and Privacy. pp. 111–125. SP '08 (2008)
8. Sweeney, L.: K-anonymity: A model for protecting privacy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems **10**(5), 557–570 (2002)