Generalization-Based Privacy-Preserving Data Collection

Lijie Zhang and Weining Zhang

Department of Computer Science, University of Texas at San Antonio
{lijez,wzhang}@cs.utsa.edu

Abstract. In privacy-preserving data mining, there is a need to consider on-line data collection applications in a client-server-to-user (CS2U) model, in which a trusted server can help clients create and disseminate anonymous data. Existing privacy-preserving data publishing (PPDP) and privacy-preserving data collection (PPDC) methods do not sufficiently address the needs of these applications. In this paper, we present a novel PPDC method that lets respondents (clients) use generalization to create anonymous data in the CS2U model. Generalization is widely used for PPDP but has not been used for PPDC. We propose a new probabilistic privacy measure to model a distribution attack and use it to define the respondent's problem (RP) for finding an optimal anonymous tuple. We show that RP is NP-hard and present a heuristic algorithm for it. Our method is compared with a number of existing PPDC and PPDP methods in experiments based on two UCI datasets and two utility measures. Preliminary results show that our method can better protect against the distribution attack and provide good balance between privacy and data utility.

1 Introduction

Consider the following scenario. A medical researcher, Steve, wants to study via data mining how cancer patients at various stages use non-prescript medicine. Due to public concerns of privacy [1], he wants to perform privacy preserving data mining (PPDM) on anonymous data. However, it is difficult to identify a source of data, since typically cancer hospitals do not document the use of non-prescript medicine and pharmacies do not document treatment of cancer patients. It is also difficult to integrate anonymous data from hospitals and pharmacies because the data do not contain personal identities. A possible solution is for Steve to use an on-line data collection service, such as a well-known survey website.

Let us consider how such a website can provide anonymous data to Steve. Perhaps the simplest solution is the privacy-preserving data publishing (PPDP) that has been extensively reported in the literature. PPDP is based on a server-to-user (S2U) model, in which one or more server (data owner such as a hospital, bank, or government agency) releases to the user (data miner) anonymous data that are obtained from the original data by perturbation [2, 3] or generalization [4, 5, 6, 7, 8, 9, 10]. Recently, generalization-based PPDP methods have gained wide acceptance and have efficient implementations [9, 11].

I.-Y. Song, J. Eder, and T.M. Nguyen (Eds.): DaWak 2008, LNCS 5182, pp. 115–124, 2008.

To provide Steve with anonymous data via PPDP, the website has to ask respondents to submit original data that contain sensitive personal information. Unlike in a typical PPDP environment, such as a hospital where patients have to provide their private information to receive services (i.e., diagnosis or treatment), on-line data collection applications rely on voluntary submission of information. We believe that not all people are willing to voluntarily submit original data all the time, even if they know that the server will not try to breach their privacy. Thus, PPDP alone does not provide a sufficient solution for the website. We also need ways for respondents to submit anonymous data.

In the literature, privacy preserving data collection (PPDC) has been proposed for respondents to submit anonymous data. PPDC is based on a client-to-user (C2U) model in which each client (respondent) submits directly to the user (data miner) an anonymous tuple that is obtained from the original tuple using perturbation [12, 13, 14]. However, existing PPDC methods have a number of problems due to the restriction of C2U model. For example, they may fail under a PCA-based attack [15] or a distribution attack (see Section 4.1).

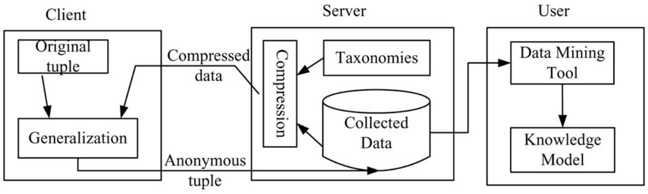


Fig. 1. PPDC in a CS2U Model

1.1 Our Contributions

To address these problems, we consider PPDC in a client-server-to-user (CS2U) model (see Figure 1) that combines S2U and C2U models. In this model, clients (respondents) create anonymous tuples with the help of a trusted server (such as the website) and submit anonymous tuples to the user (data miner) via the server. We propose a novel generalization-based PPDC method for the CS2U model. Our contributions are as follows.

1. We propose a framework for generalization-based PPDC in a CS2U model.
2. We define a new probabilistic privacy measure based on a distribution attack and use it to define the respondent's problem of finding an optimal anonymous tuple.
3. We show that this respondent's problem is NP-hard and propose a heuristic algorithm to solve it. Our generalization algorithm is able to create anonymous tuples using data already collected (rather than the entire set of original data required by existing PPDP methods) without modifying any collected tuple. To reduce the communication cost, we also present two compression methods that reduce the amount of data sent by the server to clients.

4. We compare our method with two PPDC methods and two PPDP methods in a number of experiments using two UCI datasets and two utility measures, including the information loss, which measures utility of a single anonymous tuple, and the classification accuracy of decision trees, which measures the utility of the set of collected tuples. Preliminary results show that our method balances the privacy and the utility better than existing methods do.

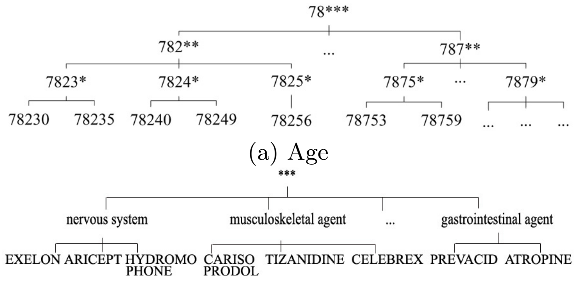
1.2 Related Work

Our work differs from well-known generalization-based PPDP methods such as k-anonymity [4, 6] and ℓ-diversity [7] in several ways. First, our algorithm only requires to access collected anonymous tuples, but their algorithms need all original tuples. Second, we consider a distribution attack, which is more general than attacks they consider. Third, we may generalize both QI and SA, but they generalize only QI.

Our work also differs from existing PPDC methods, including those based on random perturbation, such as Warner's method [12], which randomizes tuple with binary attributes, and MRR method [13], which randomizes tuples with multi-valued attributes, and those based on linear algebra, such as EPR method [14]. First, we use generalization rather than perturbation. Second, we utilizes a trusted server but they do not. Although EPR also uses a server to provide a perturbation matrix for respondents to create perturbed tuples, it distrusts the server. Third, our method protects against distribution attack, but theirs do not.

1.3 Road Map

The rest of this paper is organized as follows. In Section 2, we described a distribution attack, a privacy measure, a utility measure, and the Respondent's Problem. In Section 2.3, we show that the RP problem is NP-hard and present a heuristic algorithm to solve it. Section 4 presents preliminary results of our experiments, and Section 5 concludes the paper.



(a) Age

(b) Medication

Fig. 2. Attribute taxonomies

2 Respondent's Problem

Data tuples have a set $\mathcal{A}$ of attributes, where each attribute A has a finite set of values and a taxonomy T_A in the form of a tree (see Figure 2 for taxonomies of some attributes). Each taxonomy defines a relation $\succ$ (*more general than*) and a relation $\succeq$ (*covers*) over the values. For two values v_1 and v_2, $v_1 \succ v_2$ if v_1 is an ancestor of v_2; and $v_1 \succeq v_2$ if $v_1 \succ v_2$ or $v_1 = v_2$. We call a tuple a *base tuple* if its components are leaf values and let $\mathcal{B}$ be the set of base tuples. Both $\succ$ and $\succeq$ can be extended to tuples and components of tuples.

In our framework (see Figure 1), there is one collector (server) and a finite number of respondents (clients). Each respondent has one original tuple (which is a base tuple) and will independently create (and submit to the collector) at most one anonymous tuple. We emphasize that anonymous tuples can be submitted in any order. The collector provides respondents the set of anonymous tuples already collected. The adversary can be a respondent or the user, but not the collector.

Name	Age	Zipcode	Industry	Medication
Mike	[26,30]	78240	Univ.	EXELON
Andrew	[31,35]	78256	Electronics	Celebrex
Denny	[26,30]	78249	Univ.	atropine
Amy	[31,35]	78230	Furniture	EXELON
Angela	[51,55]	78249	Residential	Celebrex
Leo	[31,35]	78256	Furniture	Celebrex
Mark	[31,35]	78235	Furniture	atropine

(a) Original tuples

Age	Zipcode	Industry	Medication
[26,30]	7824*	Univ.	nervous
[31,35]	78256	Retail	muscloskeletal
[26,30]	7824*	Education	atropine
[31,35]	7823*	Furniture	nervous
[51,55]	78249	Building	muscloskeletal
[31,35]	78256	Furniture	muscloskeletal
[31,35]	7823*	Retail	atropine

(b) Anonymous Tuples

Fig. 3. Original vs Anonymous Data

Example 1. In Figure 3, original tuples of a set of respondents are listed in table (a) for convenience. Table (a) itself does not exist in the CS2U model. Table (b) lists anonymous tuples collected by the server at the end of PPDC process. These tuples are obtained using the algorithm described in Section 3.2. Suppose anonymous tuples are created and received in the same order as tuples are listed in table (a), the anonymous tuple of Denny (the third tuple of table (b)) will be created based on the first two tuples of table (b), that of Amy based on the first three tuples of table (b), and so on.

2.1 Linking Probability

Tuples are in the form of <QI, SA>, where QI (quasi-identifier) is a set of attributes, such as Age and Zipcode, which if joined with some publicly available database, such as the voter registry, may reveal identities of some respondents; and SA (sensitive attributes) is a set of attributes[1], such as Medication, that

[1] Without loss of generality, we assume that SA is a single attribute.

needs to be protected. Let O be a set of original tuples[2,3] and o be a random tuple from O. The probability for o to be $<q, a>$ is $p(q, a) = \frac{f_{q,a}}{|O|}$, where $f_{q,a}$ is the frequency of $<q, a>$ in O.

The adversary is interested in the conditional probability $Pr[SA = a|QI = q]$ (or simply $Pr[a|q]$), which measures the strength of a link between (the QI value of) a respondent and an SA value. Obviously, $Pr[a|q] = \frac{p(q,a)}{p(q)} = \frac{f_{q,a}}{f_q}$ and $f_q = \sum_{a \in \mathcal{B}[SA]} f_{q,a}$. Since respondents do not submit original tuples, the adversary does not know the probabilities of original tuples. However, with knowledge about a set G of collected anonymous tuples, taxonomies and value distributions of attributes, the adversary can estimate the linking probability $Pr[a|o[QI]]$ of any respondent o for any base SA value a. Specifically, the adversary can identify a *anonymous group*, which is a set of anonymous tuples whose QI values cover $o[QI]$, i.e., $G_o = \{t \mid t \in G, t[QI] \succeq o[QI]\}$, and then estimate $Pr[a|o[QI]]$ as follows.

$$Pr[a \mid o[QI], G] = \frac{\hat{f}_{o[QI],a}}{\hat{f}_{o[QI]}} = \frac{\sum_{t \in G_o} \phi_{QI}(o, t) \cdot \phi_{SA}(a, t[SA])}{\sum_{t \in G_o} \phi_{QI}(o, t)} \tag{1}$$

where $\hat{f}_{o[QI],a}$ and $\hat{f}_{o[QI]}$ estimate $f_{o[QI],a}$ and $f_{o[QI]}$, respectively. To emphasize that the estimates are based on the collected data, G is listed as a condition. We call this the *distribution attack,* which can succeed if the probability distribution is severely skewed.

We now explain Eq. (1). Since tuples in the anonymous group cover the $f_{o[QI]}$ occurrences of tuple $<o[QI], *>$, we can imagine spreading $f_{o[QI]}$ units of support of $<o[QI], *>$ over tuples in G_o. Each tuple in G_o contributes a fraction of a unit that is inversely proportional to the number of base tuples it covers. Thus, we have the following estimate.

$$\hat{f}_{o[QI]} = \sum_{t \in G_o} \left(\prod_{A \in QI} \frac{l(o[A], t[A])}{l(t[A])} \right) = \sum_{t \in G_o} \phi_{QI}(o, t) \tag{2}$$

Here, a weight can be assigned to each leaf value in taxonomies according to value distributions of attributes, and $l(v)$ (resp. $l(v, v')$) is the total weight of leaf nodes of node v (resp. leaf nodes shared by nodes v and v'). Intuitively, $\phi_{QI}(o, t)$ is the fraction contributed by t . Similarly, we also have

$$\hat{f}_{o[QI],a} = \sum_{t \in G_o} \frac{l(a, t[SA])}{l(t[SA])} \prod_{A \in QI} \frac{l(o[A], t[A])}{l(t[A])} = \sum_{t \in G_o} \phi_{SA}(a, t[SA]) \phi_{QI}(o, t) \tag{3}$$

Definition 1. *Let τ be a privacy threshold, o be a respondent (who has not yet submitted a tuple), G be the set of tuples collected (prior to o's submission), and t be the anonymous tuple from o. We say that (the submission of) t* protects *o if $Pv(o, G \cup \{t\}) \geq \tau$, where $Pv(o, G') = 1 - \max_{a \in \mathcal{B}[SA]}\{Pr[a \mid o[QI], G']\}$ is the* privacy of o with respect to dataset G'.

[2] For convenience, we use respondent and original tuple interchangeably.

[3] Since no personal identity is available, all datasets are multi-sets with possibly duplicate tuples.

2.2 Loss of Information

To find optimal anonymous tuples for respondents, we use a generic measure of utility of a tuple: the *loss of information of a tuple* t, defined as $L(t) = \sum_{A \in \mathcal{A}} L(t[A])$, where $L(v)$ is the *loss of information* of a value v. There are many ways to measure the *loss of information of a value* v. For example, $L(v) = \frac{I(v)-1}{I(r_A)}$, where $I(v)$ is the total number of leaves of nodes v in the taxonomy and r_A is the root of the taxonomy.

2.3 Problem Statement

Let o be a respondent, G be the set of anonymous tuples collected before o submits a tuple, and τ be a privacy threshold. The Respondent's Problem (RP) is to find a generalized tuple t, such that, $t = \text{argmin}_{t' \succeq o}\{L(t')\}$, subject to $Pv(o, G \cup \{t\}) \geq \tau$.

3 Analysis and Solution

3.1 Theoretical Results

The following theorem characterizes a possible solution (i.e., an anonymous tuple that protects the respondent). Due to space limit, proofs are omitted here, but they can be found in a full paper.

Theorem 1. *An anonymous tuple* $t \succeq o$ *can protect respondent* o *if and only if* $\phi_{QI}(o,t)[\phi_{SA}(a,t[SA]) - (1-\tau)] \leq M_a$, $\forall a \in \mathcal{B}[SA]$, *where* $M_a = (1-\tau)\sum_{t' \in G_o} \phi_{QI}(o,t') - \sum_{t' \in G_o}[\phi_{QI}(o,t') \cdot \phi_{SA}(a,t'[SA])]$ *is the* margin of protection *wrt* o *and* a.

Given a set of collected tuples and an SA value v, both M_a and $\phi_{SA}(a,v)$ are fully determined for every base SA value a. We can partition base SA values into 8 subsets depending on whether each of M_a and $\phi_{SA}(a,v) - (1-\tau)$ is equal to, less than, or greater than 0. Each subset (D_i) imposes a lower bound (D_i^-) and an upper bound (D_i^+) on $\phi_{QI}(o,t)$. We use these bounds to determine the existence of a solution to RP that has SA value v.

Corollary 1. *A generalized tuple with a given SA value* $v \succeq o[SA]$ *can protect respondent* o *iff* $m_1 = 0 < \phi_{QI}(o,t) \leq m_2$ *or* $0 < m_1 \leq \phi_{QI}(o,t) \leq m_2$, *where* $m_1 = \max\{D_i^- \mid i = 1, \ldots, 8\}$ *and* $m_2 = \min\{D_i^+ \mid i = 1, \ldots, 8\}$.

Thus, we can partition general tuples that cover o into subsets based on their SA values and test each partition according to Corollary 1. For each partition that contains a solution, we must solve the following RP-FixSA (i.e., RP with fixed SA value) problem to find an optimal solution tuple in the partition.

$$\begin{aligned} t = \text{argmin}_{t'}\{L(t') \mid t'[SA] = v, t' \succeq o\} \text{ subject to} \\ \phi_{QI}(o,t)(\phi_{SA}(a,v) + \tau - 1) \leq M_a, \forall a \in \mathcal{B}[SA] \text{ such that } v \succeq a \end{aligned} \tag{4}$$

However, this problem is *NP-hard* (in terms of the number of attributes and the size of attribute taxonomies). Consequently, RP is also *NP-hard*.

Theorem 2. *RP-FixSA is NP-hard.*

Corollary 2. *The Respondent's Problem is NP-hard.*

3.2 Create Anonymous Tuples

The above analysis leads to the algorithm in Figure 4, which solves RP heuristically at the client-side by first finding anonymous tuples that are possible solutions, and then choosing one of these tuples that minimizes information loss. Specifically, steps 6-7 test whether a partition may contain a solution according to Corollary 1 and findOptimalQIValue() in step 8 solves RP-FixSA.

```
Input: a private tuple $o$, a privacy threshold $\tau$, the set $G$ of collected tuples
Output: tuple $t \succeq o$ that satisfies $Pv(o, G \cup \{t\}) \geq \tau$ and minimizes $L(t)$
Method:
1.  compute anonymous group $G_o$;
2.  $T = \phi$;
3.  for each base SA value $a$, compute margin of protection $M_a$;
4.  for each SA value $v$ in the path from $o[SA]$ to root do
5.     $t[SA] = v$;
6.     compute $m_1$ and $m_2$;
7.     if ($m_1 = 0$ and $m_1 < m_2$) or ($m_1 > 0$ and $m_1 \leq m_2$)
8.        $t$ =findOptimalQIValue($t$, $o$, $G_o$, $m_2$);
9.        if $t \neq null$
10.          if $\phi_{QI}(o, t) < m_1$
11.             $t$ =specializeQI($m_1$);
12.          if $m_1 \leq \phi_{QI}(o, t) \leq m_2$
13.             $T = T \cup t$;
14. $t = \text{argmin}_{t' \in T}\{L(t')\}$;
15. return $t$;
```

Fig. 4. Algorithm: Generalize a Tuple for a Respondent (GTR)

3.3 Compress Collected Data

On the server-side, we reduce the communication cost by compressing the collected data with a simple incremental compression method. We represent each group of duplicate tuples as a tuple-count pair (t, c_t). By encoding attribute values, unique tuples are represented as numeric vectors with a total order. The tuple in each tuple-count pair is represented either by its index in this total order or by a compact binary form, giving two compression methods.

4 Experiments

We implemented five algorithms: the GTR algorithm, two existing PPDC methods: multi-variant random response (MRR) [13] and eigenvector perturbation

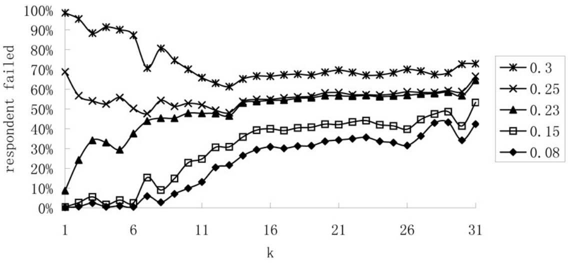
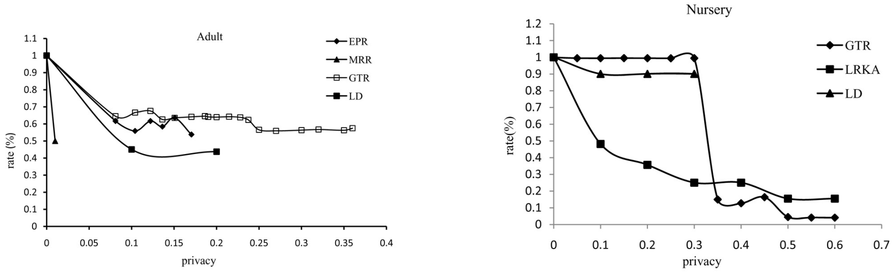


Fig. 5. Percentage of Respondents Not Protected by EPR (Adult dataset)

response (EPR) [14], and two well-known PPDP methods: Local Encoding k-anonymity (LRKA) [16] and ℓ-diversity (LD)[7]. We used two datasets from the UCI Machining Learning Repository [17]: the Adult and the Nursery, and two utility measures: information loss and classification accuracy of decision trees (learned using the ID3 decision tree mining algorithm [18]) in our experiments.

4.1 Protection against Distribution Attack

In this experiment, we investigated whether existing PPDC methods can protect against the distribution attack. Figure 5 shows the percentage of respondents (based on Adult dataset) who are not protected by EPR under a distribution attack (i.e, the privacy does not satisfy $Pv(o, G) \geq \tau$). For most values of τ, EPR fails to protect all respondents no matter what value its own privacy parameter k^* is. For example, for $\tau = 0.3$, EPR can protect at most 39% of respondents for $k^* = 31$, and less if k^* is smaller (despite that smaller k^* is thought to improve the privacy for EPR). If $k^* = 1$, EPR can protect only 0.1% of respondents. Similarly, MRR (not shown here) can only protect 40 to 60% of respondents. As a baseline of comparison, GTR always protects 100% of respondents for all values of τ, because it will not submit any data for a respondent if that person's privacy cannot be guaranteed.



(a) Adult Dataset (b) Nursery Dataset

Fig. 6. Decision Tree Accuracy of Anonymous Data

4.2 Utility of Collected Data

In this experiment, we compared utility of data obtained by different methods under comparable privacy guarantees. In Figure 6, privacy (x-axis) is measured by $Pv(o, G) \geq \tau$. We only show ranges of τ in which SA are not generalized to the root. LRKA does not appear in (a) because its results do not satisfy any τ no matter what the value of k is. For the same reason, EPR and MRR do not appear in (b). The utility (y-axis) is measured by the relative classification accuracy, that is the accuracy of decision trees learned from anonymous data divided by accuracy of decision trees learned from the original data. As shown in Figure 6, GTR outperforms these existing methods when $\tau \leq 0.36$. When $\tau > 0.4$, all methods produce very poor data because anonymous tuples are too general. The results for information loss are similar and omitted due to space limit.

5 Conclusions

In this paper, we present a novel PPDC method that lets respondents (clients) use generalization to create anonymous data in a CS2U model. Generalization has not been used for PPDC. We propose a new probabilistic privacy measure to model a distribution attack and use it to define the respondent's problem (RP) for finding an optimal anonymous tuple. We show that RP is NP-hard and present a heuristic algorithm for it. Our method is compared with a number of existing PPDC and PPDP methods in a number of experiments based on two UCI datasets and two utility measures. Preliminary results show that our method can better protect against the distribution attack and provide good balance between privacy and data utility.

Acknowledgement

The authors wish to thank the five anonymous reviewers for their constructive comments, which helped us to improve the quality of the paper. The work of Weining Zhang was supported in part by NSF grant IIS-0524612.

References

1. Cranor, L. (ed.): Communication of ACM. Special Issue on Internet Privacy vol. 42(2) (1999)
2. Agrawal, R., Srikant, R.: Privacy-preserving data mining. In: ACM SIGMOD International Conference on Management of Data, pp. 439–450. ACM, New York (2000)
3. Evfimievski, A., Gehrke, J., Srikant, R.: Limiting privacy breaching in privacy preserving data mining. In: ACM Symposium on Principles of Database Systems, pp. 211–222. ACM, New York (2003)

4. Samarati, P., Sweeney, L.: Protecting privacy when disclosing information:k-anonymity and its enforcement through generalization and suppression. In: Proc. of the IEEE Symposium on Research in Security and Privacy (1998)
5. Aggarwal, C.C.: On k-anonymity and the curse of dimensionality. In: International Conference on Very Large Data Bases, pp. 901–909 (2005)
6. Yang, Z., Zhong, S., Wright, R.N.: Anonymity-preserving data collection. In: International Conference on Knowledge Discovery and Data Mining, pp. 334–343 (2005)
7. Machanavajjhala, A., Gehrke, J., Kifer, D., Venkitasubramaniam, M.: ℓ-diversity: Privacy beyond k-anonymity. In: IEEE International Conference on Data Engineering (2006)
8. Li, N., Li, T.: t-closeness: Privacy beyond k-anonymity and l-diversity. In: IEEE International Conference on Data Engineering (2007)
9. LeFevre, K., DeWitt, D.J., Ramakrishnan, R.: Mondrian multidimensional k-anonymity. In: IEEE International Conference on Data Engineering (2006)
10. LeFevre, K., DeWitt, D.J., Ramakrishnan, R.: Incognito: Efficient fulldomain k-anonymity. In: ACM SIGMOD International Conference on Management of Data (2005)
11. Ghinita, G., Karras, P., Kalnis, P., Mamoulis, N.: Fast data anonymization with low information loss. In: International Conference on Very Large Data Bases, pp. 758–769 (2007)
12. Warner, S.L.: Randomized response: A survey technique for eliminating evasive answer bias. Journal of American Statistical Association 57, 622–627 (1965)
13. Du, W., Zhan, Z.: Using randomized response techniques for privacy-preserving data mining. In: International Conference on Knowledge Discovery and Data Mining (2003)
14. Zhang, N., Wang, S., Zhao, W.: A new scheme on privacy-preserving data classification. In: International Conference on Knowledge Discovery and Data Mining, pp. 374–382 (2005)
15. Huang, Z., Du, W., Chen, B.: Deriving private informaiton from randomized data. In: ACM SIGMOD International Conference on Management of Data, pp. 37–47 (2005)
16. Du, Y., Xia, T., Tao, Y., Zhang, D., Zhu, F.: On multidimensional k-anonymity with local recoding generalization. In: IEEE International Conference on Data Engineering (2007)
17. The uci machine learning repository, `http://mlearn.ics.uci.edu/MLRepository.html`
18. Han, J., Kamber, M.: Data Mining Concepts and Techniques. Elsevier, Amsterdam (2006)