A Hybrid Multi-Group Privacy-Preserving Approach for Building Decision Trees[⋆]

Zhouxuan Teng and Wenliang Du

Department of Electrical Engineering and Computer Science
Syracuse University, Syracuse, NY 13244, USA
Email: zhteng@syr.edu, wedu@ecs.syr.edu

Abstract. In this paper, we study the privacy-preserving decision tree building problem on vertically partitioned data. We made two contributions. First, we propose a novel *hybrid* approach, which takes advantage of the strength of the two existing approaches, randomization and the secure multi-party computation (SMC), to balance the accuracy and efficiency constraints. Compared to these two existing approaches, our proposed approach can achieve much better accuracy than randomization approach and much reduced computation cost than SMC approach.
We also propose a multi-group scheme that makes it flexible for data miners to control the balance between data mining accuracy and privacy. We partition attributes into groups, and develop a scheme to conduct group-based randomization to achieve better data mining accuracy. We have implemented and evaluated the proposed schemes for the ID3 decision tree algorithm.

Keywords: Privacy, SMC, Randomization.

1 Introduction

In today's information age, both the volume and complexity of data available for decision-making, trend analysis and other uses continue to increase. To "mine" these vast datasets for useful purposes has spurred the development of a variety of data mining techniques. Of considerable interest is abstracting information from a dataset composed of information which may be located at different sites, or owned by different people or agencies, i.e., distributed databases. However, data owners must be willing to share all their data. Issues of privacy and confidentiality can arise which prohibit data owners from contributing to a data warehouse. To address these critical privacy and confidentiality issues, privacy-preserving data mining (PPDM) techniques have emerged.

[⋆] This work was supported by Grant ISS-0219560, ISS-0312366 and CNS-0430252 from the United States National Science Foundation.

In this paper, we study a specific PPDM problem: building decision trees on vertically partitioned data sets. In this PPDM problem, the original data set D is vertically divided into two parts, with one part D_a known by Alice, and the other part D_b known by Bob. The problem is to find out how Alice and Bob conduct data mining on the vertically joint data set $D = D_a \cup D_b$, without compromising their private information.

A number of solutions have been proposed in the literature to solve various privacy-preserving data mining problems. They can be classified into two general categories: the *secure multi-party computation* (SMC) and the *randomization* approaches. In the SMC approach, Alice and Bob run a cryptographic protocol to conduct the joint computation. SMC can conduct the required computation while ensuring that the private inputs from either party are protected from each other. Previous results using the SMC approach include [3, 6, 8]. In the randomization approach, one of the parties (e.g. Alice) adds some noise to her data to disguise the original data D_a, and then she sends the disguised data set $\widehat{D_a}$ to Bob; Several schemes have been proposed for conducting data mining based on the partially disguised joint data formed by $\widehat{D_a}$ and D_b, including [2, 1, 5, 4][1].

The contribution of this paper is two-fold: First, we have developed a hybrid scheme that can harness the strength of both SMC and randomization schemes to achieve a better accuracy and efficiency. Second, we have developed a general multi-group scheme, which provides a flexible mechanism for data miner to adjust the balance between privacy and accuracy.

Our proposed hybrid approach and multi-group approach are general and can be applied to various data mining computations, including decision tree building and association rule mining. In this paper, they are applied to the ID3 decision tree algorithm[2].

2 Problem Definition and Background

In this paper we focus on a specific decision tree building problem for vertically partitioned data. The problem is illustrated in Figure 1(a).

Definition 1. *(Two-party decision tree building over vertically partitioned data) Two parties, Bob and Alice, each have values of different attributes of a data set. They want to build a decision tree based on the joint database. However neither of them wants to disclose the accurate values of the attribute he/she is holding to other party, i.e., nobody can actually have the "joint" database.*

[1] Some of these studies are not targeted at the vertically partitioned data, they can nevertheless be trivially extended to deal with this kind of data partition scenario.

[2] Our scheme can also be applied to other decision tree algorithms

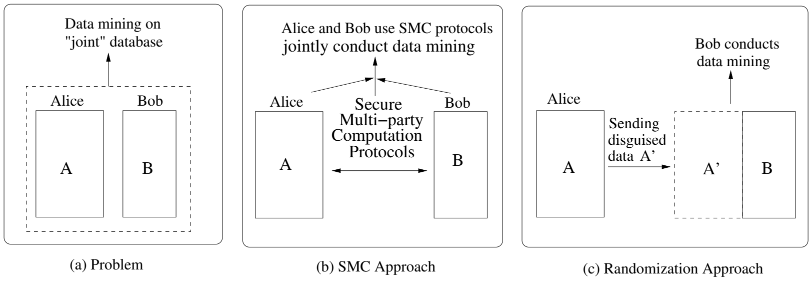


Fig. 1. Problem and different approaches.

2.1 ID3 Algorithm.

In a decision tree, each non-leaf node contains a splitting point, and the main task for building a decision tree is to identify the test attribute for each splitting point. The ID3 algorithm uses the information gain to select the test attribute. Information gain can be computed using *entropy.* In the following, we assume there are m classes in the whole training data set. We know

$$Entropy(S) = -\sum_{j=1}^{m} Q_j(S) \log Q_j(S), \tag{1}$$

where $Q_j(S)$ is the relative frequency of class j in S. We can compute the information gain for any candidate attribute A being used to partition S:

$$Gain(S, A) = Entropy(S) - \sum_{v \in A} (\frac{|S_v|}{|S|} Entropy(S_v)), \tag{2}$$

where v represents any possible values of attribute A; S_v is the subset of S for which attribute A has value v; $|S|$ is the number of elements in S.

In decision tree building, assume that the set S is associated with a node V in the tree. All the records in S has the same values for certain attributes (each corresponds to a node from the root to V). We use an logical `AND` expression $E(S)$ to encode those attributes, namely all the records in S satisfy the expression $E(S)$. Let D represent the entire data set. We use $N(E)$ to represent the number of records in the data set D that satisfies the expression E. Then,

$$\begin{aligned}
|S| &= N(E(S)) \\
|S_v| &= N(E(S_v)) \\
&= N(E(S) \wedge (A = v)) \\
Q_j(S) &= \frac{N(E(S) \wedge (Class = j))}{N(E(S))}.
\end{aligned}$$

From the above equations, we know that as long as we can compute $N(E)$ for any logical AND expression E, we can get all the elements that allow us to compute entropies and information gains. We show how to compute $N(E)$ using the SMC approach or the randomization approach for vertically-partitioned data.

The SMC Approach. The SMC approach is depicted in Figure 1(b). Let us divide E into two parts, $E = E_a \wedge E_b$, where E_a contains only the attributes from Alice, while E_b contains only the attributes from Bob. Let V_a be a vector of size n: $V_a(i) = 1$ if the ith record satisfies E_a; $V_a(i) = 0$ otherwise. Because E_a belongs to Alice, Alice can compute V_a from her own share of attributes. Similarly, let V_b be a vector of size n: $V_b(i) = 1$ if the ith data item satisfies E_b; $V_b(i) = 0$ otherwise. Bob can compute V_b from his own share of attributes.

Note that a nonzero entry of $V = V_a \wedge V_b$ (i.e. $V(i) = V_a(i) \wedge V_b(i)$ for $i = 1, \ldots, n$) means the corresponding record satisfies both E_a and E_b, thus satisfying E. To compute $N(E)$, we just need to find out how many entries in V are non-zero. This is equivalent to computing the dot product of V_a and V_b:

$$N(E) = N(E_a \wedge E_b) = V_a \cdot V_b = \sum_{i=1}^{n} V_a(i) * V_b(i).$$

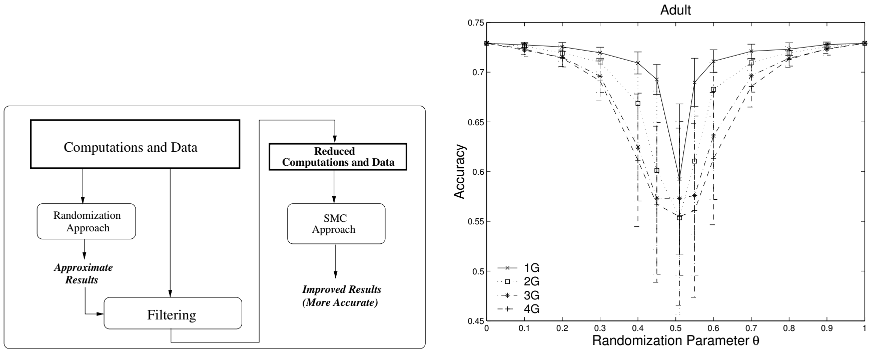
A number of dot-product protocols have already been proposed in the literature [6, 3]. With these SMC protocols, Alice and Bob can get (and only get) the result of $N(E)$, neither of them knows anything about the other party's private inputs, except the information that can be derived from $N(E)$.

The Randomization Approach. To use the randomization approach to build decision trees, Alice generates a disguised data set $\widehat{D_a}$ from her private data D_a. Alice then sends $\widehat{D_a}$ to Bob. Bob now has the full data set $\widehat{D_a} \cup D_b$, though part of which is disguised. Bob can conduct data mining based on this partially disguised data set. This approach is depicted in Figure 1(c).

There are a number of ways to perform randomization. Our scheme in this paper is based on the randomized response technique [7]. They were proposed in several existing work [5, 4] to deal with categorical data in privacy-preserving data mining. Readers can get details from the literature and we do not describe them in detail here due to page limitations.

3 A Hybrid Approach for Privacy-Preserving Data Mining

Many data mining computations involve *searching* among a set of candidates. For example, in building decision trees, at each tree node, we search for the best test attribute from a candidate set based on certain criteria; in association rule mining, we search through a set of candidates to find those whose supports are above certain threshold. Using SMC to conduct these searches is expensive since the search space can be quite large. If we can reduce the search space using some



(a) General Hybrid Approach

(b) Adult: Accuracy vs. Window Size

Fig. 2. General Hybrid Approach and Experiment Results for Adult

light-weight computations (in terms of both communication and computation costs), we can significantly reduce the total costs.

Randomization scheme is a very good choice for such a light-weight computation because of two reasons: it is much less expensive than SMC, and yet it produces results good enough for filtering purposes. If Z_u is a significant portion of Z, the costs of SMC is substantially reduced compared to the computations that use SMC alone. The entire hybrid approach is depicted in Figure 2(a).

In the next section, we describe a *Hybrid-ID3* algorithm that uses randomization to get some candidate splitting attributes at each node and then use SMC method to choose the best one from these candidates.

3.1 The *Hybrid-ID3* Algorithm.

Let D represent the entire data set. Let D_a represent the part of the data owned by Alice, and let D_b represent the part of the data owned by Bob. Alice disguises D_a using the randomization approach, and generate the disguised data set $\widehat{D_a}$; she sends $\widehat{D_a}$ to Bob. Bob does the same and sends his disguised part $\widehat{D_b}$ to Alice. Alice forms the entire data set $D_1 = D_a \cup \widehat{D_b}$, while Bob forms $D_2 = \widehat{D_a} \cup D_b$.

We describe the *Hybrid-ID3* algorithm which uses the randomization and SMC schemes as building blocks. In this algorithm, we use $N(E)$ to represent the *actual* number of records in D that satisfy the expression E (computed using the SMC approach.) We use AL to represent a set of candidate attributes. Before conducting this algorithm, Alice and Bob have already exchanged the disguised data. Namely Alice has $D_1 = D_a \cup \widehat{D_b}$, and Bob has $D_2 = \widehat{D_a} \cup D_b$.

Hybrid-ID3(E, AL)

1. Create a node V.

2. **If** $N(E \wedge (class = C)) == N(E)$ for any class C, **then** return V as a leaf node labeled with class C. Namely, all the records that satisfy E belong to class C.
3. **If** AL is empty, **then** return V as a leaf-node with the class $C = \text{argmax}_C N(E \wedge (class = C))$. Namely, C is the majority class among the records that satisfy E.
4. Find the splitting attribute using the following procedure:
   (a) For each test attribute $A \in AL$, Alice computes (estimates) A's information gain from D_1, and Bob computes A's information gain from D_2, both using the randomization approach. Alice and Bob use the average of their results as A's estimated information gain.
   (b) Select ω test attributes that have the ω highest information gains.
   (c) Using SMC to compute the actual information gains for these ω attributes, and select the one TA with the highest information gain.
5. Label node V with TA.
6. **For** each known value a_i of TA
   (a) Grow a branch from node V for the condition $TA = a_i$.
   (b) **If** $N(E \wedge (TA = a_i)) == 0$ **then** attach a leaf labeled with $C = \text{argmax}_C N(E \wedge (class = C))$, i.e., C is the majority class among the records that satisfy E.
   (c) **Else** attach the node returned by **Hybrid-ID3(**$E \wedge (TA = a_i)$**,** $AL - TA$**)**.

Note that the values of $N(E \wedge (class = C))$ at Step 2 and Step 3 can be obtained from Step 4.c of the previous round. Similarly, computations at Step 6.b can also be obtained from Step 4.c of the same round. Therefore, there are no extra SMC computations in Step 2, 3, and 6.b.

3.2 Privacy and Cost Analysis.

Because SMC computations do not reveal any more information about the private inputs than what can be derived from the results, the primary source of information disclosure is from the disguised data due to the randomization approach. Several privacy analysis methods for the randomization approach have been proposed in the literature [1, 5]. We will not repeat them in this paper.

Regarding the computation and communication costs, we are only interested in the relative costs compared to the SMC-only approach. Since the computation and the communication costs of the randomization part is negligible compared to the SMC part, we use the amount of SMC computations conducted in the hybrid approach as the measure of the cost, and we compare this cost with the amount of SMC computations conducted in the SMC-only approach. This cost ratio between these two approaches is primarily decided by the window size. We will give the simulation results in section 5.

4 The Multi-group Randomization Scheme

For many data mining computations, calculating the accurate relationship among attributes is important. Randomization tends to make this calculation less accurate, especially when each attribute is randomized independently, because of the bias introduced by the randomization schemes. The randomization schemes proposed in the literature mostly randomize attributes independently. We have found out that such randomization schemes lead to undesirable results for privacy-preserving decision tree building. To achieve better accuracy, we propose a general *multi-group* framework, which can be used for randomization schemes.

In this scheme, attributes are divided into g $(1 \leq g \leq t)$ groups (where t is the total number of attributes in the data set); randomization is applied on the unit of groups, rather than on the unit of single attribute. For example, if randomization is to add random noise, then we will add the same noise to the attributes within each group[3]. However, these numbers are independent from group to group. The advantage of this multi-group scheme is that by adding the same random noise to hide several attributes together, the relationship of these attributes are better preserved than if independent random numbers are added. However, the disadvantage of this approach is that if adversaries know the information about one attribute, they can find the information about the other attributes in the same group. Thus, there is a balance between privacy and data mining accuracy. By choosing the appropriate value of g, we can achieve a balance that is suitable for a specific application.

To demonstrate the effectiveness of this multi-group framework, we apply it to a specific randomization scheme, the randomized response scheme, which has been used by various researchers to achieve privacy-preserving data mining. We call our scheme the *Multi-group Randomized Response (MRR)* scheme. The existing randomized response schemes are special case of the MRR scheme: the scheme proposed in [4] is a 1-group scheme, while the schemes proposed in [5] are essentially t-group scheme because each attribute forms its own group.

Data Disguise. In the general randomized response technique, before sending a record to another party (or to the server), a user flips a biased coin for each attribute independently, and decides whether to tell a truth or a lie about the attribute based on the coin-flipping result. In MRR scheme, the process is still the same, the only difference is that now the coin-flipping is conducted for each group, and a user either tells a truth for all the attributes in the same group or tells a lie about all of them.

Estimating $N(E)$***.*** Let $P(E)$ represent the portion of the data set that satisfies E. Estimating $N(E)$ is equivalent to estimating $P(E)$.

Assume that the expression E contains attributes from m groups. We rewrite E using the following expression, with e_k being an expression consisting of only attributes from the group k (we call e_k a sub-pattern of E):

[3] If the domains of attributes are different, the range of the random numbers can be adjusted to match their domains.

$$E = e_1 \wedge e_2 \wedge \cdots \wedge e_m = \bigwedge_{k=1}^{m} e_k$$

We define a variation of E as $E' = f_1 \wedge \cdots \wedge f_m$, where f_i is equal to either e_i or the bitwise-opposite of e_i (i.e. $\overline{e_i}$). For each expression E, there are totally 2^m different variations, including E itself. We denote these variations of E as E_0 to E_ω, where $E_0 = E$ and $\omega = 2^m - 1$.

Theorem 1. *Let $P(E_i \to E_j)$ represent the probability that an expression E_i in the original data becomes an expression E_j in the disguised data after the randomized response process. We have the following formula:*

$$P(E_i \to E_j) = \theta^u (1-\theta)^{m-u},$$

where u represents the number of the common bits between the binary forms of number i and number j.

Proof. Proof is omitted due to page limitations.

Let $P^*(E)$ represent the expected number of records, in the *disguised* data set, that satisfies the expression E. $P^*(E)$ can be estimated by counting the number of records that satisfy E in the *disguised* data set. Obviously, we have

$$P^*(E) = \sum_{i=0}^{\omega} P(E_i \to E_j) P(E_i)$$

If we define a matrix A, such that $A(i,j) = P(E_i \to E_j)$ for $i = 0, \cdots, \omega$ and $j = 0, \cdots, \omega$, we get the following linear system of equations.

$$\begin{pmatrix} P^*(E_0) \\ \vdots \\ P^*(E_\omega) \end{pmatrix} = A \begin{pmatrix} P(E_0) \\ \vdots \\ P(E_\omega) \end{pmatrix}$$

Theorem 2. *The matrix A defined as above is invertible if and only if $\theta \neq 0.5$.*

Proof. Proof by induction and the proof is omitted due to page limitations.

In situations where $P(E)$ is the only thing we need, just like in the ID3 decision tree building algorithm, there is a much more efficient solution with cost $O(m)$ instead of $O(2^m)$. This technique is similar to the one used in [5] and it is omitted here due to page limitations.

5 Evaluation

To evaluate the proposed hybrid scheme, we have selected three databases from the UCI Machine Learning Repository [4]: *Adult*, *Mushroom*, and *Tic-tac-toe* datasets. We randomly divide all attributes of each data set into two parts with the same cardinality: Alice and Bob's share respectively.

In our experiments, we always used 80% of the records as the training data and the other 20% as the testing data. We use the training data to build the decision trees, and then use the testing data to measure how accurate these trees can predict the class labels. The percentage of the correct predictions is the *accuracy* value in our figures. We repeat each experiment for multiple times, and each time the disguised data set is randomly generated from the same original data set. We plot the means and the standard deviation for the accuracy values. The results for *Tic-tac-toe* dataset is omitted due to page limitations.

5.1 Accuracy vs. number of groups.

Figure 2(b) shows the change of accuracy along the number of groups in the *randomization-only* approach for *Adult* dataset. In the figure, "1G", "2G", "3G", and "4G" indicate that the data are disguised using the 1-group, 2-group, 3-group, and 4-group randomization schemes respectively. From the figure, we can see that the accuracy decreases when the number of groups increases. When θ is close to 0.5 (e.g., $\theta = 0.4$), the rate of deterioration is rapid as the number of group increases. It is interesting to see that the results of the 4-group scheme are very close to those of the 3-group scheme. This is because in this specific *Adult* dataset, most of the expressions that are evaluated in building the tree contain attributes from less than 3 groups.

5.2 Accuracy: Hybrid vs. Randomization-Only.

Figures 3(a) and 4(a) show the accuracy comparisons between the hybrid approach and the randomization-only approach. The vertical bars in the figures depict the standard deviations. The comparisons are shown for different randomization parameter θ and for different window size ω. In these three figures, "4G" and "1G" indicate that the data are disguised using the 4-group randomization scheme and the 1-group randomized scheme, respectively.

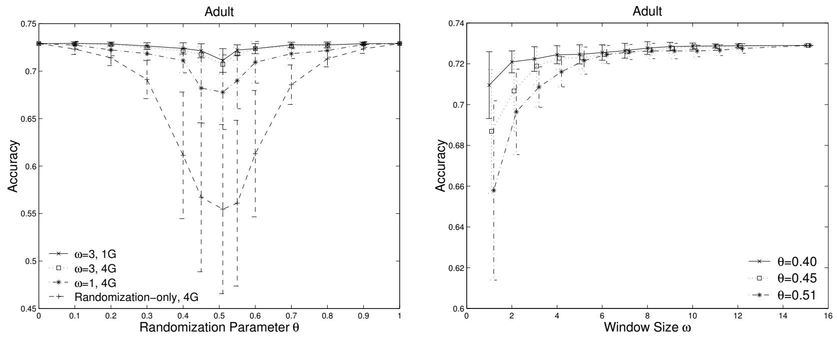
The figures clearly show that the hybrid approach achieves a significant improvement on accuracy compared to the randomization-only approach. When θ is near 0.5, the accuracy of the trees built via the randomization-only approach is just slightly better than the random guess (a random guess can yield 50% of accuracy on average). In contrast, the trees built via the hybrid approach can achieve a much better accuracy.

When the window size is increased to 3, the accuracy difference between the 4-group randomization scheme and the 1-group randomization scheme becomes

[4] ftp://ftp.ics.uci.edu/pub/machine-learning-databases

much small. This means, choosing the 4-group randomization scheme does not degrade the accuracy much when $\omega = 3$, while at the same time, it achieves better privacy than the 1-group randomization scheme.

A surprising result in all these three figures is that when the window size is set to 1, the accuracy can be improved significantly compared to the randomization-only approach. Initially we thought that the hybrid approach with $\omega = 1$ is equivalent to the randomization-only approach. From this result, we realized that they are different, and the difference is at Step 2 and 6.b of the **Hybrid-ID3** algorithm. Step 2 detects whether all the records associated with the current tree node belong to a single class C. If so, we will not further split this node. With the hybrid approach, such a detection is conducted using SMC, which always generates the accurate results. However, using the randomization-only approach, because the result is inaccurate, it is very likely that we will continue splitting the node even when such a splitting is unnecessary. These extra splittings may result in a dramatic different tree structure compared to the tree built upon the original undisguised data, thus cause the significant difference in their accuracy results. Step 6.b has the similar effect.



(a) Hybrid vs. Randomization-Only

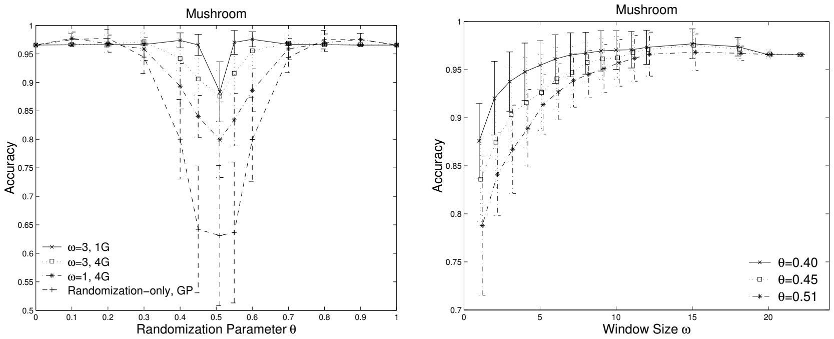
(b) Accuracy vs. Window Size

Fig. 3. Experiment Results for Adult Data Sets

5.3 Accuracy vs. Window Size ω.

Figures 3(b) and 4(b) show the relationship between the accuracy and the window size in the hybrid approach where the number of groups g is 4.

The figures show that increasing SMC window size increases the accuracy of the decision tree. The increase is quite rapid when the window size is small; after certain point, the change of the window size does not affect the accuracy much. This means that the actual best test attribute is very likely among the top few



(a) Hybrid vs. Randomization-Only

(b) Accuracy vs. Window Size

Fig. 4. Experiment Results for Mushroom Data Sets

candidates. This indicates that choosing a small window size can be the very cost-effective: it achieves a decent degree of accuracy without having to conduct many expensive SMC computations.

5.4 Efficiency Improvement.

The motivation of the hybrid approach is to achieve better accuracy than the randomization-only approach, as well as achieve better efficiency than the SMC-only approach. Our previous experiments have shown the accuracy improvement. We now show how well the hybrid approach achieves the efficiency goal. We have summarized the efficiency improvement in Table 1, alone with the degree of accuracy achieved (4-group randomization and $\theta = 0.45$).

In Table 1, A is the accuracy of the hybrid approach minus the accuracy of the randomization-only approach, C is the ratio of the total number of SMC computations in the hybrid approach to that in the SMC-only approach.

The table shows that the efficiency improvement for the Mushroom data set is the most significant. This is because the number of attributes in the Mushroom data set is larger. This trend indicates that the larger the number of attributes, the higher level of efficiency improvement.

6 Conclusions and Future Work

We have described a hybrid approach and a multi-group randomization approach for privacy-preserving decision tree buildings over vertically-partitioned data. The hybrid approach combines the strength of the SMC approach and the randomization approach to achieve both high accuracy and efficiency. Our experiments show that the hybrid approach achieves significantly better accuracy compared to the randomization-only approach and it is much more efficient than the SMC-only approach. Our multi-group randomization approach allows data miners to control the trade-off between privacy and data mining accuracy.

For the hybrid approach, we only used a fixed window size throughout the entire decision tree building process. In the future, we will investigate whether a dynamic window size can help further improve the performance, i.e., the window size for different tree nodes might be different, depending on the randomization results. We will also investigate the effectiveness of the hybrid approach on other data mining computations.

Table 1. Performance Improvement

	$\omega = 2$		$\omega = 3$		$\omega = 4$	
	A	C	A	C	A	C
Adult	0.14	19%	0.15	28%	0.16	37%
Mushroom	0.23	10%	0.26	15%	0.27	20%

References

1. D. Agrawal and C. Aggarwal. On the design and quantification of privacy preserving data mining algorithms. In *Proceedings of the 20th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, Santa Barbara, California, USA, May 21-23 2001.
2. R. Agrawal and R. Srikant. Privacy-preserving data mining. In *Proceedings of the 2000 ACM SIGMOD on Management of Data*, pages 439–450, Dallas, TX USA, May 15 - 18 2000.
3. W. Du and Z. Zhan. Building decision tree classifier on private data. In *Workshop on Privacy, Security, and Data Mining at The 2002 IEEE International Conference on Data Mining (ICDM'02)*, Maebashi City, Japan, December 9 2002.
4. W. Du and Z. Zhan. Using randomized response techniques for privacy-preserving data mining. In *Proceedings of The 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 505–510, Washington, DC, USA, August 24-27 2003.
5. A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke. Privacy preserving mining of association rules. In *Proceedings of 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Edmonton, Alberta, Canada, July 2002.
6. J. Vaidya and C. Clifton. Privacy preserving association rule mining in vertically partitioned data. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, July 23-26 2002.
7. S. L. Warner. Randomized response: A survey technique for eliminating evasive answer bias. *Journal of the American Statistical Association*, 60(309):63–69, March 1965.
8. Z. Yang, S. Zhong, and R.N. Wright. Anonymity-preserving data collection. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Chicago, IL, USA., August 21-24 2005.