Privacy-Sensitive Bayesian Network Parameter Learning

D. Meng and K. Sivakumar
School of EECS, Washington State University
Pullman, WA 99164-2752, USA
{dmeng, siva}@eecs.wsu.edu

H. Kargupta*
Department of CSEE, UMBC
Baltimore, MD 21250, USA
hillol@cs.umbc.edu

Abstract

This paper considers the problem of learning the parameters of a Bayesian Network, assuming the structure of the network is given, from a privacy-sensitive dataset that is distributed between multiple parties. For a binary-valued dataset, we show that the count information required to estimate the conditional probabilities in a Bayesian network can be obtained as a solution to a set of linear equations involving some inner product between the relevant different feature vectors. We consider a random projection-based method that was proposed elsewhere to securely compute the inner product (with a modified implementation of that method).

1. Introduction

Advances in networking, storage, and computing technologies have resulted in an unprecedented increase in the amount of data collected and made available to the public. This explosive growth in digital data has brought increased concerns about the privacy of personal information [1]. Privacy is also an important issue in applications related to counter-terrorism and homeland security. For example, mining healthcare data for the detection of bio-terrorism may require mining clinical records and pharmaceutical purchases of certain specific drugs. However, combining such diverse datasets belonging to different parties may violate privacy laws. Therefore, it is important to be able to extract desired data mining models from the data, without accessing the raw data in its original form.

Privacy-sensitive data mining is an evolving area within the broad field of data mining [2, 3, 8]. In the following, we briefly review some of the important approaches proposed in the literature. Due to space constraints, we cite only a few important works.

*Also affiliated with AGNIK, LLC, USA.

1.1. Related Work

There exists a growing body of literature on privacy-sensitive data mining. These algorithms can be divided into two broad groups: (a) approaches based on randomization and (b) approaches based on secure multi-party computation (SMC).

The first approach to privacy-sensitive data mining starts by first perturbing the data using randomized techniques. The perturbed data is then used to extract the patterns and models. The randomized value distortion technique for learning decision trees [2] is an example of this approach. See [7] for a possible privacy breach using this approach. Evfimievski et al. [5] have also considered the approach in [2] in the context of association rule mining and suggest techniques for limiting privacy breaches.

SMC is the problem of evaluating a function of two or more parties' secret inputs, such that each party finally learns their specified function output and nothing else is revealed, except what is implied by the party's own inputs and outputs. SMC problem was first introduced by Yao [11]. Du and Atallah have presented a collection of new secure multi-party computation applications such as privacy-sensitive statistical analysis [4]. Clifton [3] has described several secure multi-party computation based algorithms that can support privacy-sensitive data mining. Feigenbaum et al. have addressed the problem of computing approximations using SMC [6]. More recently, Wright and Yang [10] have proposed a privacy-sensitive Bayesian Network structure learning algorithm.

1.2. Our Contribution

In this paper, we consider the problem of learning the parameters of a Bayesian Network (BN), assuming the structure of the network is given, from a privacy-sensitive dataset that is distributed between multiple parties. For a binary-valued dataset, we show that the

count information required to estimate the conditional probabilities (model parameters) in a Bayesian network can be obtained as a solution to a set of linear equations involving some inner product between the relevant different feature vectors. Therefore, any privacy-sensitive method for computing inner product between vectors can be used to solve the Bayesian network parameter learning problem. Specifically, we consider a random projection-based method (to compute the inner product) that was proposed elsewhere [9].

The rest of the paper is organized as follows. Section 2 provides a brief overview of Bayesian Networks (BN) followed by a description of the problem statement. In Section 3, we describe our proposed algorithm. Experimental results are presented in Section 4. Section 5 concludes the paper.

2. Problem Description

A BN is a probabilistic graph model, which is an important tool in data mining. It can be defined as a pair $(\mathcal{G}, p)$, where $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a directed acyclic graph (DAG). For a variable $X \in \mathcal{V}$, a parent of X is a node from which there exists a directed link to X. Figure 1 is a BN called the ASIA model. Let $pa(X)$ denote the set of parents of X, then the conditional independence property can be used to factor the joint probability as follows: $P(\mathcal{V}) = \prod_{X \in \mathcal{V}} P(X \mid pa(X))$. The set of conditional distributions $\{P(X \mid pa(X)), X \in \mathcal{V}\}$ are called the parameters of a Bayesian network. Learning a BN involves learning the structure of the network and obtaining the conditional probabilities associated with the network.

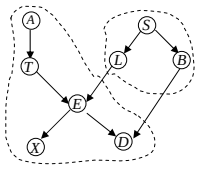

Figure 1. ASIA Model

We consider a set-up where the data corresponding to the different nodes are distributed among two or more parties. For example, in the ASIA model of Figure 1, party I contains observations for features (nodes) A, T, E, X, and D, whereas party II contains observations for features S, L, and B. This is usually referred to vertical partitioning of the data or a heterogeneous data distribution. The dataset is privacy-sensitive in the sense that each party does not wish to share its raw data with the other parties. However, they wish to mine the combined dataset to obtain a global BN. We assume that the structure of the global BN is known to all the parties and focus on the problem of estimating the parameters of the network. Our proposed solution to this problem is presented in the following section.

3. Algorithm

In the following, we assume that the features of the BN are binary, taking values in the set $\{-1, 1\}$. Extensions to multi-variate (discrete) case is conceptually similar, except for the algebra. In Section 3.1, we describe a system of linear equations, whose solution yields the desired conditional probabilities. The coefficient matrix for the linear equations can be obtained from the BN structure, which is assumed to be known. The inner product between certain feature vectors are needed to obtain the "right-hand-side vector" of the linear equations. Any secure inner product computation module can be used for this purpose. This is discussed in Section 3.2. Finally, Section 3.3 provides a privacy analysis of the proposed method.

3.1. Equations for BN parameter learning

In this section we build a set of linear equations whose solution yields all the conditional probabilities for a BN. We assume that all the data are binary with values 1 or -1 and the structure of BN is given.

For simplicity, first consider a node z with two parent nodes x and y. We need to obtain the values of all the conditional probabilities for z, given the values of nodes x and y. As shown in Table 1, there are eight $(2^3 = 8)$ different count values — $\{a, b, \ldots, h\}$ — to be determined. For example, b represents the number of observations with $x = -1$, $y = 1$ and $z = -1$. The corresponding probabilities can be obtained simply by normalizing the count values with respect to the total number of observations N.

Let ${N_{ijk}}^{xyz}$ denote the number of observations for which $x = i$, $y = j$, and $z = k$, for $i, j, k \in \{-1, 1\}$. We then have $P(z = k \mid x = i, y = j) = \frac{{N_{ijk}}^{xyz}}{{N_{ij}}^{xy}}$, $i, j, k \in \{-1, 1\}$, where ${N_{ij}}^{xy}$ denotes the number of observations for which $x = i$, and $y = j$.

Definition 3.1 *(Pseudo inner product) Given $n \geq 1$ vectors $x_1, x_2, \ldots, x_n$, each of dimension k, we define their pseudo-inner product (pip) $pip(x_1, x_2, \ldots, x_n) = \sum_{j=1}^{k} \prod_{i=1}^{n} x_{ij}$, where $x_i = [x_{i1}, x_{i2}, \ldots, x_{ik}]$, $i = 1, 2, \ldots, n$ are the components of vector x_i.*

Let N be the total number of observations and X, Y, Z denote the data vector (column vector) for nodes x, y, z, respectively. Since there are three data

Table 1. Three-node example

	x, y			
	$-1, -1$	$-1, 1$	$1, -1$	$1, 1$
$z = -1$	a	b	c	d
$z = 1$	e	f	g	h

vectors, we can compute $2^3 - 1 = 7$ different pseudo inner products. Observe that each pseudo inner product can be expressed uniquely by count variables $a, b, \ldots, h$. For example, $pip(Z)$ equals the sum of the entries in vector Z, which is precisely the number of observations with $z = 1$ minus the number of observations with $z = -1$. Indeed, we can write $(e + f + g + h) - (a + b + c + d) = pip(Z)$. Another obvious condition is: $(e+f+g+h)+(a+b+c+d) = N$. Indeed, we can write eight linear equations as follows: $Ax = b$, where

$$A = \begin{pmatrix} -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ -1 & -1 & 1 & 1 & -1 & -1 & 1 & 1 \\ -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix},$$

$x = [a, b, c, d, e, f, g, h]^T$, and $b = [pip(Z),\ pip(X),\ pip(Y),\ pip(Z, X),\ pip(Z, Y),\ pip(Z, X, Y),\ pip(X, Y),\ N]^T$. It is easy to verify that matrix A is nonsingular. So we can solve the linear equations to get all the required conditional probabilities.

This simple idea can be easily generalized to the case of arbitrary number of parent nodes. The proof is by induction and has been omitted due to page limitations (see `http://www.eecs.wsu.edu/~siva/icdm_04_longversion.pdf` for details).

3.2. Secure Inner Product Computation

From the previous subsection, we know if a BN structure is given, the coefficient matrix A is uniquely determined. Therefore, if we can compute the pseudo inner products, the BN parameters can be obtained by solving the linear equations. If the variables corresponding to the parent node(s) of a given node belong to a different party than the variable of the node itself, then computing the pseudo inner product would require exchange of raw data between the parties. Therefore, we need a privacy-sensitive method to compute inner products in order to accomplish this step.

In our experiments, we used a random projection based method proposed in [9]. The important equations are reproduced below: Let U be an $m \times n$ data matrix, with m observations and n features. Suppose R is an $m \times m$ orthogonal matrix; i.e., $R^T R = RR^T = I$. Consider the (multiplicatively) perturbed matrix $U_1 = RU$. Note that we use a single projection as opposed to the proposed double projection in [9]. It is easy to see that $U_1^T U_1 = (U^T R^T)(RU) = U^T (R^T R)U = U^T U$. Therefore the inner products between the columns of U can be computed using the perturbed matrix U_1. So the owner of the data set U computes U_1 and hands over that to the other party (or a third party who does the data mining), who can then compute the required pseudo inner products. In practice, perturbation matrix R is chosen to be a random orthogonal matrix. This can be accomplished by starting with a random matrix with independent identically distributed (i.i.d.) entries W and orthogonalizing it.

3.3. Communication, Error, and Privacy Analysis

We now present a brief analysis of the communication cost and privacy of the proposed scheme.

First observe that for those nodes, all of whose parents are in the same site, there is no privacy or communication problem and those parameters (conditional probabilities) can be locally estimated and communicated to the other parties.

Suppose, node i has $n_a - 1$ parents at the same site and n_b parents at a different site. Therefore, roughly $2^{n_a} 2^{n_b} = 2^{n_a + n_b}$ pseudo inner products have to be computed securely. This would require communication of $O(m2^{n_i})$ bits, where $n_i = n_a + n_b$ is one more than the number of parents of node i. Therefore, the total communication cost is $O(m \sum_i 2^{n_i})$. Note that in typical BN applications $n_i << n$.

The pseudo inner product computation is the only step that requires some exchange of data between the parties. Therefore, any privacy breach would have to occur in that step. Theorem 1 in [9] discusses the privacy preserving properties of the random projection method. In particular, the $m \times m$ random orthogonal matrix R has $m(m-1)/2$ independent random entries (the rest of the $m(m+1)/2$ entries being determined by orthogonality constraints). As such, there are infinitely many solutions U, in general, to $U_1 = RU$, if R is unknown. By using a single random orthogonal matrix R in the projection instead of two random matrices R_1, R_2 as in [9], we do not have to "average" over results over multiple trials. Moreover, inner products computed using a single random orthogonal matrix R are virtually error-free as opposed to the case with double projection using random matrices, where the error goes to zero as the number of independent trials goes

to infinity. More details about the privacy preserving properties of single and double projection methods can be found in [9].

4. Experimental Results

In this section, we present results of our experiments with the proposed privacy-sensitive BN parameter learning for the ASIA model (see Figure 1). The true conditional probabilities (parameters) of the ASIA model for nodes E and D (nodes with parents from different sites) are given in Table 2. A data set with 2000 samples was generated from this model.

Table 2. True conditional probabilities

E	0.9	0.1	0.1	0.01	0.1	0.9	0.9	0.99
D	0.9	0.2	0.8	0.9	0.1	0.8	0.2	0.1

We generated a random matrix R_1 whose entries were i.i.d. Gaussian with zero mean and unit variance. This matrix was then orthogonalized using a QR decomposition to obtain a random orthogonal matrix R. The estimated parameters using our proposed algorithm in Section 3 are tabulated in Table 3. As expected, the estimated parameters are almost identical to the true values.

Table 3. Estimated conditional probabilities

E	0.9	0.1	0.1	0.01	0.1	0.9	0.9	0.99
D	0.89	0.21	0.81	0.84	0.11	0.79	0.19	0.16

5. Discussion and Conclusions

We considered the problem of learning the parameters of a Bayesian Network, assuming the structure of the network is given, from a privacy-sensitive dataset that is distributed between multiple parties. We considered the case of vertical (or heterogeneous) partitioning, where different parties hold values corresponding to a different subset of the variables. For a binary-valued dataset, we showed that the count information required to estimate the conditional probabilities of a Bayesian network can be obtained as a solution to a set of linear equations involving some inner product between the relevant different feature vectors. In our experiments, we considered a random projection-based method with a single projection using a random orthogonal matrix. This implementation requires considerably less exchange of perturbed data and produces almost error-free results as compared with that using double projection using random matrices.

Acknowledgements

The authors acknowledge supports from the United States National Science Foundation grants IIS-0329143 and IIS-0350533.

References

[1] The end of privacy. The Economist, May 1999.

[2] R. Agrawal and S. Ramakrishnan. Privacy-preserving data mining. In *Proceedings of SIGMOD Conference*, pages 439–450, 2000.

[3] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, and M. Zhu. Tools for Privacy Preserving Distributed Data Mining. *ACM SIGKDD Explorations*, 4(2):28–34, 2003.

[4] W. Du, Y. S. Han, and S. Chen. Privacy-preserving multivariate statistical analysis: Linear regression and classification. In *Proceedings of 2004 SIAM International Conference on Data Mining (SDM04)*, pages 222–233, Lake Buena Vista, FL, April 2004.

[5] A. Evfimevski, J. Gehrke, and R. Srikant. Limiting privacy breaches in privacy preserving data mining. In *Proceedings of the ACM SIGMOD/PODS Conference*, pages 211–222, San Diego, CA, June 2003.

[6] J. Feigenbaum, Y. Ishai, T. Malkin, K. Nissim, M. Strauss, and R. Wright. Secure multiparty computation of approximations. In *Proceedings of the 28th International Colloquium on Automata, Languages, and Programming (ICALP)*, volume 2076 of *Lecture Notes in Computer Science*, pages 927–938, Berlin, 2001. Springer.

[7] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar. On the Privacy Preserving Properties of Random Data Perturbation Techniques. In *Proceedings of the IEEE International Conference on Data Mining*, pages 99–106, Melbourne, FL, November 2003.

[8] Y. Lindell and B. Pinkas. Privacy preserving data mining. In *Advances in Cryptology - CRYPTO*, pages 36–54, 2000.

[9] K. Liu, H. Kargupta, and J. Ryan. Multiplicative noise, random projection, and privacy preserving data mining from distributed multi-party data. Technical Report TR-CS-03-24, Computer Science and Electrical Engineering Department, University of Maryland, Baltimore County, 2003.

[10] R. Wright and Z. Yang. Privacy-preserving Bayesian network structure computation on distributed heterogeneous data. In *Proceedings of the tenth ACM SIGKDD Conference*, Seattle, WA, August 2004.

[11] A. C. Yao. How to generate and exchange secrets. In *Proceedings 27th IEEE Symposium on Foundations of Computer Science*, pages 162–167, 1986.