Privacy-Preserving Naive Bayesian Classification

Zhijun Zhan
Department of Computer Science
Syracuse University, USA
email: zhzhan@ecs.syr.edu

LiWu Chang
Center for High Assurance Computer Systems
Naval Research Laboratory, USA
email: lchang@itd.nrl.navy.mil

Stan Matwin
School of Information Technology & Engineering
University of Ottawa, Canada
email: stan@site.uottawa.ca

ABSTRACT
Privacy is an important issue in data mining and knowledge discovery. In this paper, we propose to use the randomized response techniques to conduct the data mining computation. Specifically, we present a method to build naive Bayesian classifiers from the disguised data. We conduct experiments to compare the accuracy of our classifier with the one built from the original undisguised data. Our results show that although the data are disguised, our method can still achieve fairly high accuracy.

KEY WORDS
Privacy, security, naive Bayesian classification, data mining

1 Introduction

Data mining has emerged as a means for identifying patterns and trends from a large amount of data [7]. To conduct data mining computations, we need to collect data first. Without privacy concerns, data can be directly collected. However, because of privacy concerns, some people might decide to selectively divulge information, or give false information, or simply refuse to disclose any information at all. A survey was conducted in 1999 [3] to understand Internet users' attitudes towards privacy. The result shows 17% of respondents are privacy fundamentalists, who are extremely concerned about any use of their data and generally unwilling to provide their data, even when privacy protection measures were in place. However, 56% of respondents are a pragmatic majority, who are also concerned about data use, but are less concerned than the fundamentalists; their concerns are often significantly reduced by the presence of privacy protection measures. The remaining 27% are marginally concerned and are generally willing to provide data under almost any condition, although they often expressed a mild general concern about privacy. According to this survey, providing privacy protection measures is a key to the success of data collection. *How can we improve the chance to collect more truthful data that are useful for data mining while preserving users' privacy? How can users contribute their personal information without compromising their privacy?*

One way to achieve privacy is to use anonymous techniques [1], which allow users to disclose their personal information without disclosing their identities. The biggest problem of using anonymous techniques is that there is no guarantee on the quality of the data set. A malicious user (e.g., a competing company) could send a great deal of random information to the database and render the database useless, or a company could send a lot of made-up information to the database with the goal of making their products the most favorable ones. These potential attacks could all render the database useless. If the communication is really anonymous, it is difficult for the database owner to control the quality of the data. To guarantee the quality, it is important for the database owner to verify the identities of the data contributors.

Another way to achieve privacy is to let each user disguise or randomize their data, such that the data collector cannot derive the truthful information about an user's private information. The challenge is how to conduct data mining from the disguised data? To address this challenge, we first propose the following computing model: The model consists of a data collection step and a computation step. In the data collection step, each user utilizes certain techniques to disguise his/her data, then sends the disguised data to the central warehouse; the central warehouse should not be able to find out any user's actual data with probabilities better than a pre-defined threshold. In the computation step, the central warehouse constructs a database using the disguised data, and conducts data mining computations on this database. The goal of the central warehouse is to derive useful information (or knowledge) out of this disguised database. In this paper, we particularly focus on a specific data mining computation, the naive Bayesian (e.g., NB) based classification [8]. The basic idea of NB classification is to construct a NB network, which is a very simple Bayesian network with an assumption that every

variable (feature) of the data is independent given the class label, to conduct the classification.

We propose to use the *Randomized Response* techniques to solve the privacy-preserving data mining problem. The basic idea of randomized response is to scramble the data in such a way that the central warehouse cannot tell with probabilities better than a pre-defined threshold whether the data from a customer contain truthful information or false information. Although information from each individual user is scrambled, if the number of users is significantly large, the aggregate information of these users can be estimated with decent accuracy. Such property is useful for naive Bayesian based classification since it is based on aggregate values of a data set, rather than individual data items.

The contributions of this paper are as follows: (1) We have modified the naive Bayesian classification algorithm [8] to make it work with data modified by randomized response techniques, and implemented the modified algorithm. (2) We then conducted a series of experiments to measure the accuracy of our modified naive Bayesian algorithm on randomized data. Our results show that if we choose the appropriate randomization parameters, the accuracy we have achieved is very close to the accuracy achieved using the original naive Bayesian classification on the original data.

The rest of the paper is organized as follows: we discuss related work in Section 2. In Section 3, we describe how to utilize multivariate randomized response technique to build naive Bayesian classifier on randomized data. In Section 4, we describe our experimental results. We give our conclusion in Section 5.

2 Related Work

Agrawal and Srikant proposed a scheme for privacy-preserving data mining using random perturbation [2]. In their scheme, a random number is added to the value of a sensitive attribute. For example, if x_i is the value of a sensitive attribute, $x_i + r$, rather than x_i, will appear in the database, where r is a random value drawn from some distribution. The paper shows that if the random number is generated with some known distribution (e.g., uniform or Gaussian distribution), it is possible to recover the distribution of the values of that sensitive attribute. Assuming independence of the attributes, the paper then shows that a decision tree classifier can be built with the knowledge of distribution of each attribute.

Rizvi and Haritsa presented a scheme called **MASK** to mine associations with secrecy constraints in [10]. Evfimievski et al. proposed an approach to conduct privacy preserving association rule mining based on randomization techniques [6]. Du and Zhan [5] utilized randomized response technique for decision tree classification.

There are currently two approaches to achieve privacy-preserving data mining: one is the perturbation approach which we had discussed in the above. The other approach is to use Secure Multi-party Computation (SMC) techniques [14]. Several SMC-based privacy-preserving data mining schemes have been proposed [15, 4, 9, 12]. These studies mainly focused on two-party distributed computing, and each party usually contributes a set of records. Although some of the solutions can be extended to solve our problem (n party problem), the performance is not desirable when n becomes large.

3 Building Naive Bayesian Classifiers Using Multivariate Randomized Response Techniques

Randomized Response techniques were first introduced by Warner [13] in 1965 as a technique to solve the following survey problem: to estimate the percentage of people in a population that has attribute A, queries are sent to a group of people. Since the attribute A is related to some confidential aspects of human life, respondents may decide not to reply at all or to reply with incorrect answers.

To enhance the level of cooperation, instead of asking each respondent whether he/she has attribute A, the interviewer asks each respondent two related questions, the answers to which are opposite to each other [13]. For example, the questions could be like the following. If the statement is correct, the respondent answers "yes"; otherwise he/she answers "no".

1. I have the sensitive attribute A.
2. I do not have the sensitive attribute A.

Respondents use a randomizing device to decide which question to answer, without letting the interviewer know which question is answered. The randomizing device is designed in such a way that the probability of choosing the first question is θ, and the probability of choosing the second question is $1-\theta$. Although the interviewer learns the responses (e.g., "yes" or "no"), he/she does not know which question was answered by the respondents. Thus the respondents' privacy is preserved. Since the interviewer's interest is to get the answer to the first question, and the answer to the second question is exactly the opposite to the answer for the first one, if the respondent chooses to answer the first question, we say that he/she is telling the truth; if the respondent chooses to answer the second question, we say that he/she is telling a lie.

To estimate the percentage of people who has the attribute A, we have

$$P^*(A = yes) = P(A = yes) \cdot \theta + P(A = no) \cdot (1 - \theta)$$
$$P^*(A = no) = P(A = no) \cdot \theta + P(A = yes) \cdot (1 - \theta),$$

where $P^*(A = yes)$ (resp. $P^*(A = no)$) is the proportion of the "yes" (resp. "no") responses obtained from the survey data, and $P(A = yes)$ (resp. $P(A = no)$) is the estimated proportion of the "yes" (resp. "no") responses to the sensitive questions. Getting $P(A = yes)$ and $P(A = no)$ is the goal of the survey. By solving the above equations, we can get $P(A = yes)$ and $P(A = no)$ if $\theta \neq \frac{1}{2}$.

The randomized response technique discussed above considers only one attribute. However, in data mining, data sets usually consist of multiple attributes; finding the relationship among these attributes is one of the major goals for data mining. Therefore, we need the randomized response techniques that can handle multiple attributes while supporting various data mining computations. Work has been proposed to deal with surveys that contain multiple questions [11]. However, their solutions can only handle very low dimensional situation (e.g., dimension = 2), and cannot be extended to solve data mining problems, in which the number of dimensions is usually high. We have developed a multivariate randomized response technique (MRR) to deal with multiple attributes.

3.1 Notations

In this work, we assume data are binary, but the techniques can be extended to categorical data. Suppose there are N attributes (A_1, ..., A_N) in a data set. Let E represent any logical expression based on those attributes (e.g., $E = (A_1 = 1) \wedge (A_2 = 0)$); let $\overline{E}$ denote the logical expression that reverses the 1's in E to 0's and 0's to 1's; we call $\overline{E}$ the opposite of E. For example, for the E in the previous example, $\overline{E} = (A_1 = 0) \wedge (A_2 = 1)$.

Let $P^*(E)$ be the proportion of the records in the whole *disguised* data set that satisfy $E =$ `true`. Let $P(E)$ be the proportion of the records in the whole *undisguised* data set that satisfy $E =$ `true` (the undisguised data set contains the true data, but it does not exist). $P^*(E)$ can be observed from the disguised data, but $P(E)$, the actual proportion that we are interested in, cannot be observed from the disguised data because the undisguised data set is not available to anybody; we have to estimate $P(E)$. The goal of MRR is to find a way to estimate $P(E)$ from $P^*(E)$.

In our multivariate scheme, we also divide each expression E to multiple sub-expressions. For example, in a two-group scheme, we write $E = E_1E_2$, where E_i contains only the attributes in the group i.

3.2 One-Group Scheme

In the one-group scheme, all the attributes are put in the same group, and all the attributes are either reversed together or keeping the same values. In other words, when sending the private data to the central database, users either tell the truth about all their answers to the sensitive questions or tell the lie about all their answers. The probability for the first event is θ, and the probability for the second event is $1-\theta$. For example, assume an user's truthful values for attributes A_1, A_2, and A_3 are 110. The user generates a random number from 0 to 1; if the number is less than θ, he/she sends 110 to the data collector (i.e., telling the truth); if the number is bigger than θ, he/she sends 001 to the data collector (i.e., telling lies about all the questions). Because the data collector does not know the random number generated by users, the data collector cannot know whether data provider tells the truth or a lie. To simplify our presentation, we use $P(11)$ to represent $P(A_1 = 1 \wedge A_2 = 1)$, $P(00)$ to represent $P(A_1 = 0 \wedge A_2 = 0)$ ("$\wedge$" is the logical `and` operator.).

Because the contributions to $P^*(11)$ and $P^*(00)$ partially come from $P(11)$, and partially come from $P(00)$, we can derive the following equations:

$$\begin{array}{rcl} P^*(11) & = & P(11) \cdot \theta + P(00) \cdot (1-\theta) \\ P^*(00) & = & P(00) \cdot \theta + P(11) \cdot (1-\theta) \end{array} \quad (1)$$

By solving the above equations, we can get $P(11)$, the information needed to build a naive Bayesian classifier. The general model for the one-group scheme is described in the following:

$$\begin{array}{rcl} P^*(E) & = & P(E) \cdot \theta + P(\overline{E}) \cdot (1-\theta) \\ P^*(\overline{E}) & = & P(\overline{E}) \cdot \theta + P(E) \cdot (1-\theta) \end{array} \quad (2)$$

Using the matrix form, let M_1 denote the coefficiency matrix of the above equations, and let $p = \theta$ and $q = (1-\theta)$, then

$$\begin{pmatrix} P^*(E) \\ P^*(\overline{E}) \end{pmatrix} = M_1 \begin{pmatrix} P(E) \\ P(\overline{E}) \end{pmatrix}, \quad (3)$$

where

$$M_1 = \begin{bmatrix} p & q \\ q & p \end{bmatrix}$$

3.3 Two-Group Scheme

In the one-group scheme, if the interviewer somehow knows whether the respondents tell a truth or a lie for one attribute, he/she can immediately obtain all the true values of a respondent's response for all other attributes. To improve the privacy level of data, data providers divide all the attributes into two groups (all the data providers should group the attributes in the same ways, e.g., one user lets attribute A_1 and A_2 to be in the group 1, then other users also let attribute A_1 and A_2 to be in the group 1). They then apply the randomized response techniques for each group *independently*. For example, the users can tell the truth for one group while telling the lie for the other group. With this scheme, even if the interviewers know information about one group, they will not be able to derive the information for the other group because they are disguised independently.

To show how to estimate $P(E_1E_2)$, we look at all the contributions to $P^*(E_1E_2)$. Parts that contribute to $P^*(E_1E_2)$ include not only the probability of the event that users tell the truth about all the answers for both groups (i.e., $P(E_1E_2)$), but also probabilities of all other events (i.e., $P(E_1\overline{E_2})$, $P(\overline{E_1}E_2)$, and $P(\overline{E_1E_2})$). In terms of θ, $P(E_1E_2)$, $P(E_1\overline{E_2})$, $P(\overline{E_1}E_2)$ and $P(\overline{E_1E_2})$ are respectively, θ^2, $\theta(1-\theta)$, $(1-\theta)\theta$ and $(1-\theta)^2$. We then have the following equation:

$$P^*(E_1E_2)=P(E_1E_2)\cdot\theta^2+P(E_1\overline{E_2})\cdot\theta(1-\theta)+ P(\overline{E_1}E_2)\cdot\theta(1-\theta)+P(\overline{E_1E_2})\cdot(1-\theta)^2$$

There are four unknown variables in the above equation ($P(E_1E_2)$, $P(E_1\overline{E_2})$, $P(\overline{E_1}E_2)$, $P(\overline{E_1E_2})$). To solve the above equation, we need three more equations. We can derive them using the similar method. The final equations are described in the following:

$$\begin{pmatrix} P^*(E_1E_2) \\ P^*(E_1\overline{E_2}) \\ P^*(\overline{E_1}E_2) \\ P^*(\overline{E_1E_2}) \end{pmatrix} = M_2 \cdot \begin{pmatrix} P(E_1E_2) \\ P(E_1\overline{E_2}) \\ P(\overline{E_1}E_2) \\ P(\overline{E_1E_2}) \end{pmatrix}, \quad (4)$$

where M_2 is the coefficiency matrix, and let $p=\theta$ and $q=1-\theta$, then,

$$M_2 = \begin{bmatrix} p^2 & pq & pq & q^2 \\ pq & p^2 & q^2 & pq \\ pq & q^2 & p^2 & pq \\ q^2 & pq & pq & p^2 \end{bmatrix} \quad (5)$$

Since two-group scheme is sufficient for naive Bayesian classification computations, we will not show the estimation model for the cases where the group number is greater than two.

3.4 Building Naive Bayesian Classifiers

Classification is one of the forms of data analysis that can be used to extract models describing important data classes or to predict future data. It has been studied extensively by the community in machine learning, expert system, and statistics as a possible solution to the knowledge discovery problem. Classification is a two-step process. First, a model is built given the input of training data set which is composed of data tuples described by attributes. Each tuple is assumed to belong to a predefined class described by one of the attributes, called the class label attribute. Second, the predictive accuracy of the model (or classifier) is estimated. A test set of class-labeled samples is usually applied to the model. For each test sample, the known class label is compared with predictive result of the model.

The naive Bayesian classifier is one of the most successful algorithms on many classification domains. Despite of its simplicity, it is shown to be competitive with other complex approaches especially in text categorization and content based filtering. Under a conditional independence assumption, i.e., $P(A_i,A_j|C) = P(A_i|C)P(A_j|C)$, for $1 \le i \ne j \le n$, the naive Bayesian classifier classifies a new data x into the class with the largest posterior probability as shown in Eq. 6, where A_i and A_j represent the attributes or variable, C is the class variable, n is the number of the attributes. Further, this posterior classification rule can be transformed into joint probability classification rule, since $P(A_1,A_2,\cdots,A_n)$ for a given data is a constant with regards to C. Finally, combining the independence assumption, the classification rule is changed into a decomposable form.

$$\begin{aligned} c &= argmax_{C_i} \ P(C_i|A_1,A_2,\cdots,A_n) \\ &= argmax_{C_i} \ \frac{P(C_i)*P(A_1,A_2,\cdots,A_n|C_i)}{P(A_1,A_2,\cdots,A_n)} \\ &= argmax_{C_i} \ P(C_i)*P(A_1,A_2,\cdots,A_n|C_i) \\ &= argmax_{C_i} \ P(C_i)\Pi_{j=1}^{n}P(A_j|C_i) \\ &= argmax_{C_i} \ P(C_i)\Pi_{j=1}^{n}\frac{P(A_j,C_i)}{P(C_i)} \end{aligned} \quad (6)$$

To build the NB classifier, we need to compute $P(C_i)$ and $P(A_j,C_i)$. Without loss of generality, we assume the database only contains binary values, and we will show how to compute these terms based on disguised training datasets.

Let E be a logical expression based on attributes. Let $P(E)$ be the proportion of the records in the undisguised data set (the true but non-existing data set) that satisfy $E=$ `true`. Because of the disguise, $P(E)$ cannot be observed directly from the disguised data, and it has to be estimated. Let $P^*(E)$ be the proportion of the records in the disguised data set that satisfy $E=$ `true`. $P^*(E)$ can be computed directly from the disguised data.

To compute $P(C_i)$, we can utilize one-group model (Eq. 2) with $E=C_i$ and $\overline{E}=\overline{C_i}$, and $P^*(E)$ and $P^*(\overline{E})$ can be computed directly from the (whole) disguised data set. Therefore, by solving the above equations (when $\theta \ne \frac{1}{2}$), we can get $P(E)$ which is $P(C_i)$ in this case.

To compute $P(A_j,C_i)$, we need to know whether A_j and C_i belong to the same group. If they come from the same group, we can still use estimation model (Eq. 2) with $E=(A_j \wedge C_i)$ and $\overline{E}=(\overline{A_j} \wedge \overline{C_i})$. However, if A_j and C_i belong to different groups, we need to utilize the estimation model for the two-group scheme (Eq. 4) with $E_1=A_j$, $\overline{E_1}=\overline{A_j}$, $E_2=C_i$ and $\overline{E_2}=\overline{C_i}$. Once we obtain $P(A_j,C_i)$ and $P(C_i)$, a NB classifier can be constructed.

3.5 Testing

Conducting the testing is straightforward when data are not disguised, but it is a non-trivial task when the testing data set is disguised. Imagine, when we choose a record from the testing data set, compute a predicted class label using the naive Bayesian classifier, and find out that the predicated label does not match with the record's actual label, can we say this record fails the testing? If the record is a true one, we can make that conclusion, but if the record is a false one (due to the randomization), we cannot. How can we compute the accuracy score of the NB classifier?

We also use the randomized response techniques to compute the accuracy score. For simplicity, we only describe how to conduct testing using the two-group scheme (since one-group is a special case for two-group scheme). We use an example to illustrate how we compute the accuracy score. Assume the number of attributes is 2, and the probability $\theta = 0.8$. To test a record $(A_1 = 1, A_2 = 0)$ (denoted by 10), with A_1 belonging to group 1 and A_2 belonging to group 2, we feed 10, 11, 00, 01 to the classifier. We know one of the class-label prediction result is true, but don't exactly know which one. However, with enough testing data, we can estimate the total accuracy score, even though we do not know which test case produces the correct prediction result.

Using the (disguised) testing data set $S = S_1S_2$, we construct other data sets $\overline{S_1}S_2$, $S_1\overline{S_2}$, $\overline{S_1S_2}$, by reversing the corresponding values in S_1 and S_2 (change 0 to 1 and 1 to 0). Note that each record in $\overline{S_i}$ (for $i \in [1, 2]$) is the opposite of the corresponding record in S_i. We say that $\overline{S_i}$ is the opposite of the data set S_i. Similarly, we define U_i as the *original undisguised* testing data set, and $\overline{U_i}$ as the opposite of U_i.

Let $P^*(cc)$ be the proportion of correct predictions from testing data set S_1S_2, $P^*(\overline{c}c)$ be the proportion of correct predictions from testing data set $\overline{S_1}S_2$, $\cdots$, $P^*(\overline{cc})$ be the proportion of correct predictions from testing data set $\overline{S_1S_2}$. Similarly, let $P(cc)$ be the proportion of correct predictions from the *original undisguised* data set U_1U_2, $P(\overline{c}c)$ be the proportion of correct predictions from $\overline{U_1}U_2$, $\cdots$, $P(\overline{cc})$ be the proportion of correct predictions from $\overline{U_1U_2}$. $P(cc)$ is what we want to estimate.

Because $P^*(cc)$, $P^*(\overline{c}c)$, $\cdots$ and $P^*(\overline{cc})$ consist of contributions from $P(cc)$, $P(\overline{c}c)$, $\cdots$ and $P(\overline{cc})$, we have the following formula:

$$\begin{pmatrix} P^*(cc) \\ P^*(c\overline{c}) \\ P^*(\overline{c}c) \\ P^*(\overline{cc}) \end{pmatrix} = M_2 \cdot \begin{pmatrix} P(cc) \\ P(c\overline{c}) \\ P(\overline{c}c) \\ P(\overline{cc}) \end{pmatrix} \quad (7)$$

where M_2 is defined in Eq.(4). $P^*(cc)$, $P^*(c\overline{c})$, $P^*(\overline{c}c)$ and $P^*(\overline{cc})$ can be obtained from testing data set S_1S_2, $S_1\overline{S_2}$, $\overline{S_1}S_2$ and $\overline{S_1S_2}$. By solving the above formula, we can get $P(cc)$, the accuracy score of testing.

4 Experimental Results

To evaluate the effectiveness of our multivariate randomized response techniques on naive Bayesian classifier, we compare the classification accuracy of our multivariate scheme with the original accuracy, which is defined as the accuracy of the classifier induced from the original data.

4.1 Data Setup

We conduct experiments on two real life data sets. We obtain the data sets from the UCI Machine Learning Repository(ftp://ftp.ics.uci.edu/pub/machine-learning-databases). The first dataset is called *Adult*. It contains 48842 instances with 14 attributes (6 continuous and 8 nominal) and a label describing the salary level. Prediction task is to determine whether a person's income exceeds 50k/year based on census data. We used first 10,000 instances in our experiment. The second data set is called *Breast-Cancer*. It has 699 instances with 10 attributes. Prediction task is to decide whether a person is benign or malignant.

We modified the naive Bayesian classification algorithm to handle the randomized data based on our proposed methods. We run this modified algorithm on the randomized data and obtain a classifier. We also apply the naive Bayesian classification algorithm to the original data set and obtain the other classifier. We then applied the same testing data to both classifiers. Our goal is to compare the classification accuracy of these two classifier. Obviously we want the accuracy of the classifier built based on our method to be close to the accuracy of the classifier built from the original algorithm.

4.2 Experimental Steps

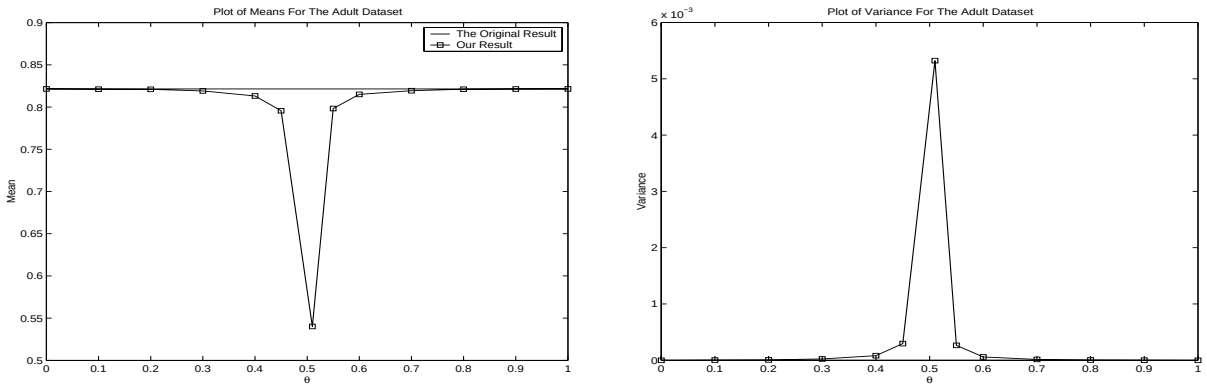
Our experiments consist of the following steps:

Preprocessing: Since we assume that the data set contains only binary data, we first transformed the original non-binary data to the binary. We split the value of each attribute from the median point of the range of the attribute. After preprocessing, we divided the data sets into a training data set D and a testing data set B. Note that B will be used for comparing our results with the benchmark results.

Benchmark: We use D and the original NB classification algorithm to build a classifier T_D; we use the data set B to test the classifier, and get an accuracy score. We call this score the original accuracy (or the benchmark score).

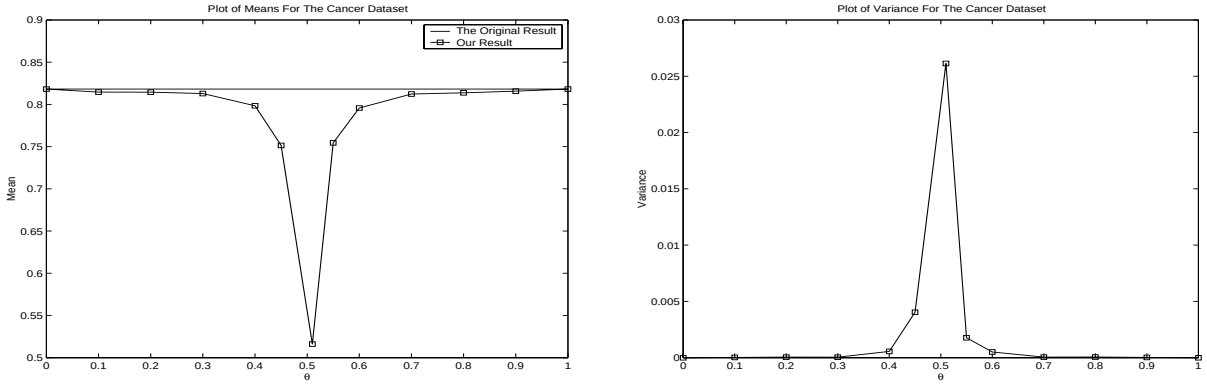
θ Selection: For $\theta = 0.0, 0.1, 0.2, 0.3, 0.4, 0.45, 0.51, 0.55$ 0.6, 0.7, 0.8, 0.9, and 1.0, we conduct the following 4 steps:

1. Randomization: We create a disguised data set G. For each record in the training data set D, we generate a random number r from 0 to 1 using uniform distribution. If $r \leq \theta$, we copy the record to G without any change; if $r > \theta$, we copy the opposite of the record to G - each attribute value of the record that we put into G is exactly the opposite of the value in the original record. We perform this randomization step for all the records in the training data set D and generate the new data set G.



(a) Mean (b) Variance

Figure 1. The Results On The Adult Data Set



(a) Mean (b) Variance

Figure 2. The Results On The Breast-Cancer Data Set

2. Classifier Construction: We use the data set G and our modified NB classification algorithm to build a naive Bayesian classifier T_G.

3. Testing: We use the data set B to test T_G, and we get an accuracy score S.

4. Repetition: We repeat steps 1-3 for 100 times, and get $S_1, \ldots, S_{100}$. We then compute the mean and the variance of these 100 accuracy scores.

4.3 The Result Analysis

4.3.1 The Analysis of Mean

Fig. 1(a) and 2(a) shows the mean values of the accuracy scores for *Adult* and *Breast-Cancer* data sets respectively. We can see from the figures that when $\theta = 1$ and $\theta = 0$, the results are exactly the same as the results when the original classification algorithm is applied. This is because when $\theta = 1$, the randomized data sets are exactly the same as the original data set D; when $\theta = 0$, the randomized data sets are exactly the opposite of the original data set D. In both cases, our algorithm produces the accurate results (comparing to the original algorithm), but privacy is not preserved in either case because an adversary can know the real values of all the records provided that he/she knows the θ value. When θ moves from 1 and 0 towards 0.5, the mean of accuracy has the trend of decreasing. When θ is around 0.5, the mean deviates a lot from the original accuracy score.

4.3.2 The Analysis of Variance

Fig. 1(b) and 2(b) shows the variances of the accuracy scores for *Adult* and *Breast-Cancer* data sets respectively. When θ moves from 1 and 0 towards 0.5, the degree of randomness in the disguised data is increased, the variance of the estimation used in our method becomes large. When the randomization level θ is different, the variance will be different. When θ is near 0.5, the randomization level is much higher and true infor-

mation about the original data set is better disguised, in other words, more information is lost; therefore the variance is much larger than the case when θ is not around 0.5. This is actually what we have predicted. We use a simple example to illustrate why this happens. Assume we have just one attribute, with 90% of 1's and 10% of 0's. If we choose $\theta = 0.5$, according to our randomization scheme, the disguised data set will contain $90\% * 0.5 + 10\% * 0.5 = 50\%$ of 1's and another 50% of 0's. If we change the distribution to 10% of 1's and 90% of 0's, we get the same results. This means when $\theta = 0.5$, information about the data distribution is lost. That is why when θ closes to 0.5 the accuracy becomes very low (Note that 0.5 is a very low accuracy, because if one just randomly guesses the class label, 1 out 2 guesses will be correct if we have just two class labels. Therefore even the random guess can achieve accuracy of 0.5.), and the variance becomes very large.

4.3.3 Summary

Our results on the two real life data sets indicate that the multivariate randomized response techniques can be utilized for privacy-preserving naive Bayesian classification. When θ is 0 or 1, which provides all the true information, the accuracy of the classifier is the highest and the privacy level of the data is the lowest. When θ is away from 0 (or 1) and approaches to 0.5, the accuracy of the classifier decreases and the privacy level of the data increases. The accuracy is dependent on the recoverability of the original data from the randomized data. The empirical results confirm that recoverability and privacy are complementary goals, and that research presented here allows a quantitative evaluation of the trade-offs between the two, e.g., if the recoverability of the original data is 20%, privacy will be at most 80%.

When $\theta = 0.5$, the related model cannot be applied, and other techniques such as randomized response techniques using the *Unrelated-Question* model may be employed. Note that In our experiment, we didn't randomize the class label and therefore only one-group scheme is implemented.

5 Concluding Remarks

In this paper, we have presented a method to build naive Bayesian classifiers using multivariate randomized response technique. The experimental results show that when we select the randomization parameter θ from $[0.6, 1]$ and $[0, 0.4]$, we can get fairly accurate classifiers comparing to the classifiers built from the undisguised data. In our future work, We will apply our techniques to solve other data mining problems (i.e., association rule mining) and extend our solution to deal with the cases where data type is not binary.

References

[1] Anonymizer.com: http://www.anonymizer.com.

[2] R. Agrawal and R. Srikant. Privacy-preserving data mining. In *Proceedings of The ACM SIGMOD Conference On Management of Data*, Dallas, Texas, USA, 2000.

[3] L. F. Cranor, J. Reagle, and M. S. Ackerman. Beyond concern: Understanding net users' attitudes about online privacy. Technical report, AT&T Labs-Research, April 1999. Available from `http://www.research.att.com/library/trs/TRs/99/99.4.3/report.htm`.

[4] 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.

[5] 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*, Washington, DC, USA, August 24-27 2003.

[6] A. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke. Privacy preserving mining of association rules. In *Proc. of 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, July 2002.

[7] J. Han and M. Kamber. *Data Mining Concepts and Techniques*. Morgan Kaufmann Publishers, 2001.

[8] Iba W. Langley, P. and K. Thompson. An analysis of Bayesian classifiers. In *Proceedings of the Tenth National Conference on Artificial Intrlligence*, pages 223–228, San Jose, CA. AAAI Press.

[9] Y. Lindell and B. Pinkas. Privacy preserving data mining. *Advances in Cryptology - CRYPTO '00*, 1880 of Lecture Notes in Computer Science. Spinger-Verlag:36–54, 2000.

[10] S. Rizvi and J.R. Haritsa. 8.shariq rizvi and jayant r. haritsa maintaining data privacy in association rule mining. In *Proceedings of the 28th VLDB Conference*, Hong Kong, China, 2002.

[11] A. C. Tamhane. Randomized response techniques for multiple sensitive attributes. *The American Statistical Association*, 76(376):916–923, December 1981.

[12] 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.

[13] S. L. Warner. Randomized response: A survey technique for eliminating evasive answer bias. *The American Statistical Association*, 60(309):63–69, March 1965.

[14] A. C. Yao. Protocols for secure computations. In *Proceedings of the 23rd Annual IEEE Symposium on Foundations of Computer Science*, 1982.

[15] Z. Zhan and L. Chang. Privacy-preserving collaborative data mining. In *Workshop on Foundation and New Direction of Data Mining at The 2003 IEEE International Conference on Data Mining (ICDM'03)*, Melbourne, Florida, USA, November 19 2003.