

# Deriving Private Information from Randomized Data

Zhengli Huang, Wenliang Du and Biao Chen  
Department of Electrical Engineering and Computer Science  
Syracuse University, Syracuse, NY 13244  
Tel: 315-443-9180 Fax: 315-443-1122  
{zhuang,wedu,bichen}@ecs.syr.edu

## ABSTRACT

Randomization has emerged as a useful technique for data disguising in privacy-preserving data mining. Its privacy properties have been studied in a number of papers. Kargupta et al. challenged the randomization schemes, and they pointed out that randomization might not be able to preserve privacy. However, it is still unclear what factors cause such a security breach, how they affect the privacy preserving property of the randomization, and what kinds of data have higher risk of disclosing their private contents even though they are randomized.

We believe that the key factor is the correlations among attributes. We propose two data reconstruction methods that are based on data correlations. One method uses the Principal Component Analysis (PCA) technique, and the other method uses the Bayes Estimate (BE) technique. We have conducted theoretical and experimental analysis on the relationship between data correlations and the amount of private information that can be disclosed based on our proposed data reconstructions schemes. Our studies have shown that when the correlations are high, the original data can be reconstructed more accurately, i.e., more private information can be disclosed.

To improve privacy, we propose a modified randomization scheme, in which we let the correlation of random noises “similar” to the original data. Our results have shown that the reconstruction accuracy of both PCA-based and BE-based schemes become worse as the similarity increases.

## Keywords

Privacy-Preserving Data Mining, Randomization, PCA, and Bayes Estimate.

## 1. INTRODUCTION

With the advance of the information age, data collection and data analysis have exploded both in size and complexity. The attempt to extract important patterns and trends from the vast data sets has led to a challenging field called

Data Mining. When a complete data set is available, various statistical, machine learning and modeling techniques can be applied to analyze the data. In many contexts, data are distributed across different sites. Traditionally, the data warehousing approach has been used to mine distributed databases. It requires that data from all the participating sites are collected at a centralized warehouse. However, many data owners may be reluctant to share their data with others due to privacy and confidentiality concerns. This is a serious impediment to perform mutually beneficial data mining tasks.

Privacy-Preserving Data Mining (PPDM) has emerged to address this issue. The research of PPDM is aimed at bridging the gap between collaborative data mining and data confidentiality. It involves many areas such as statistics, computer sciences, and social sciences. It is of fundamental importance to homeland security, modern science, and to our society in general.

Agrawal and Srikant first proposed using randomization to solve PPDM problems [2]. In their randomization 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. It is shown that given the distribution of random noises, recovering the distribution of the original data is possible. The randomization techniques have been used for a variety of privacy preserving data mining work [1, 21, 9, 7].

Kargupta et al. challenged the randomization schemes, and they pointed out that randomization might not be secure [16]. They proposed a random matrix-based Spectral Filtering (SF) technique to recover the original data from the perturbed data. Their results have shown that the recovered data can be reasonably close to the original data. The results indicate that for certain types of data, randomization might not preserve privacy as much as we have believed.

Motivated by Kargupta et al’s work, we want to answer a series of important questions that are still unanswered: *what are the key factors that decide the accuracy of the data reconstruction? what are the conditions that make data less privacy preserving using randomization? can we improve randomization to achieve better privacy?* Being able to answer these questions is important to understand how secure the randomization schemes are: first it tells us what types of data should not use the randomization to disguise; second, this understanding gives us a clear direction on how to improve the randomization to achieve better privacy preservation.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGMOD 2005 June 14-16, 2005, Baltimore, Maryland, USA  
Copyright 2005 ACM 1-59593-060-4/05/06 \$5.00.

We hypothesize that the relationships among data attributes might be the key factor that decides how much privacy can be preserved. Our hypothesis is motivated by the following intuitive extreme case: Assume that there are  $m$  numbers that have exactly the same values  $z$ . If each of them is disguised by an independent uniformly-random number with mean zero, we can estimate the value  $z$  by calculating the mean of these  $m$  perturbed numbers. As we know, the mean converges to  $z$  when  $m$  becomes large. Although the above example is unrealistic, it indicates that when data are highly correlated (thus redundant), we are able to derive, from the disguised data, more accurate information about the original data. In other words, there exists a strong relationship between the correlation and the randomization’s privacy-preserving property.

The goal of this paper is to find out such a relationship, then based on which to understand how well the randomization works in terms of privacy preserving. We have developed two methods that exploit the correlations among data to reconstruct the original data from a randomized data set.

Our first scheme is based on Principal Component Analysis (PCA) method, which provides a framework for us to control the degree of redundancy, we choose to use a scheme that is directly based on PCA theory. Kargupta’s scheme is also based on PCA, but part of it is based on Matrix perturbation theory, which makes it difficult to achieve a clear understanding of the correlation-vs-privacy relationship. The choice of directly basing on PCA is not motivated by the performance (actually, both schemes have similar performance under some conditions), it is rather motivated by its simplicity and being able to give an intuitive theoretical explanation on why it works. We call both schemes the PCA-based schemes.

We have also developed a method that is more general than the PCA-base schemes. In this scheme, we formulate the data reconstruction problem as an estimation problem, i.e., given the disguised data  $Y$ , we find  $X$ , such that the posterior probability  $P(X | Y)$  is maximized. This is exactly the Bayes estimate [20]. We use the Bayes estimate techniques to solve  $X$ , and then use  $X$  as the final reconstructed data. Our results show that this method can obtain more accurate results than the PCA-based schemes.

Based on our conclusion that correlation can reveal private information, we propose a modified randomization scheme, in which we let correlations of random noise “similar” to the original data. We have shown that the results of data reconstructions based on both PCA-based and BE-based schemes become worse when the correlation of noise becomes more and more “similar” to the original data.

The rest of the paper is organized as follows. We discuss the related work in Section 2. In Section 3, we summarize the factors that can affect the privacy of randomization. In Section 4, we show a univariate data reconstruction scheme that does not exploit data correlations. The result of this scheme is used as the baseline data for our comparison. In Section 5 and 6, we present our PCA-based data reconstruction scheme and BE-based data reconstruction scheme, respectively. The experiment results are presented in Section 7. Section 8 describes an improved randomization scheme and its experiment results. Finally we summarize our work in Section 9.

## 2. RELATED WORK

There are two general approaches to privacy preserving data mining: the randomization approach and the Secure Multi-party Computation (SMC) approach. In the randomization approach, random noises are added to the original data, and only the disguised data are shared [2, 1, 21, 9, 7]. There are two different randomization methods: the Random Perturbation scheme and the Randomized Response scheme.

Agrawal and Srikant proposed a scheme for privacy-preserving data mining using random perturbation [2]. This work has been extended by Agrawal and Aggarwal [1]. Under the scheme, Evfimievski et al. proposed an approach to conduct Privacy-Preserving Association Rule Mining [9].

The randomized response is mainly used to deal with categorical data. Rizvi and Haritsa presented a scheme called *MASK* to mine associations with secrecy constraints [21]; Du and Zhan proposed an approach to conduct Privacy-Preserving Decision Tree Building [7]. All these approaches are based on the Randomized Response technique proposed by Warner [26].

Privacy is analyzed in most of the above studies, in addition, two studies have focused on the privacy analysis. The first one is due to Evfimievski et al. [8], and the other is due to Kargupta et al. [16]. In their paper, Evfimievski et al. presented a formula of privacy breaches and a methodology to limit the breaches in the field of association rule mining.

Kargupta et al. pointed out an important issue: arbitrary randomization is not safe [16]. Inspired by their work, we study why and how correlations affect privacy. In addition to correlations, we identify other potential factors that can influence privacy.

Another approach to achieve Privacy-Preserving Data Mining is to use Secure Multi-party Computation (SMC) techniques. Briefly, an SMC problem deals with computing certain function on multiple inputs, in a distributed network where each participant holds one of the inputs; SMC ensures that no more information is revealed to a participant in the computation than what can be inferred from the participant’s input and the final output [11].

Several SMC-based privacy-preserving data mining schemes have been proposed [17, 19, 23, 24, 5]. Lindell and Pinkas used SMC to build decision trees over the horizontally partitioned data [17]. Vaidya and Clifton proposed the solutions to the clustering problem [24] and the association rule mining problem [23] for vertically partitioned data. Several SMC tools and fundamental techniques are also proposed in the literature [19, 5]. Some more schemes were presented in recent conferences as follows. Wright et al. [27] and Meng et al. [18] used SMC to solve privacy-preserving Bayesian network problems. Gilburd et al. proposed a new privacy model,  $k$ -privacy, for real-world large-scale distributed systems [10]. Sanil et al. described a privacy-preserving algorithm of computing regression coefficients [22]. Du et al. have developed building blocks to solve secure two-party Multivariate Linear Regression and Classification problems [6]. Wang et al. used an iterative bottom-up generalization to generate data, which remains useful to classification but difficult to disclose private sources [25].

### 3. DERIVING PRIVATE INFORMATION

Kargupta et al. used a data reconstruction approach to derive private information from a disguised data set [16]. Namely, a new data set  $X^*$  is reconstructed from the disguised data using certain algorithms, and the difference between  $X^*$  and the actual original data set  $X$  indicates how much private information can be disclosed. The further apart  $X^*$  is from  $X$ , the higher level of the privacy preservation is achieved. Therefore, the difference between  $X^*$  and  $X$  can be used as the measure to quantify how much privacy is preserved. Our work also uses data reconstruction approaches, but we propose two different data reconstruction algorithms.

A variety of information can lead to the disclosure of private information in a disguised data set. We summarize several of them in the following:

- **Attribute Dependency:** Attributes in many data sets are not independent, and some attributes might have a strong correlation among themselves. It is important to understand how such relationship can cause private information disclosure.
- **Sample Dependency:** For certain types of data sets, such as the time series data, there exists serial dependency among the samples. Even after perturbing the data with random noise, this dependency can still be recovered. For instance, various techniques are available from the signal processing literature to de-noise the contaminated signals. One interesting research problem is: for different types of data, what kind of dependency relationships will help the adversaries reconstruct the original data?
- **Partial Value Disclosure:** In practice, it is possible that the values of some attributes can be disclosed (via other channels). For example, assume we have a medical database that is disguised by randomization schemes. Knowing that the patient Alice has diabetes and heart problems, we might be able to estimate the other information about her. How to quantify privacy under these circumstances?
- **Data Mining Results:** In the SMC approach, all the participating parties can see the final results. These results contain aggregate information about the data, which can lead to possible privacy breaches. For example, in the association rule mining, assume that there is a rule saying that  $A$  implies  $B$  with 90% of support. Even if one party knows only  $A$  and the association rule results, he or she will be able to infer  $B$  with high confidence. How do various data mining results, including classification models, association rules, and clustering affect individual privacy? Kantarcioglu et al. has initiated studies on this issue [15].

The scope of the problems described above are broader than what we have covered in this paper. In this paper, we focus on the first problem, i.e., how to use data correlation information to derive private information?

### 4. UNIVARIATE DATA RECONSTRUCTION

In this section, we describe two data reconstruction methods derived from the existing work on randomization. The

first approach is only based on the distribution of noise. It does not consider the distribution of the original data  $X$ . The second approach bases its guess on the posterior distribution  $P(X|Y)$ , which can be estimated from the disguised data. Because both reconstruction methods treat each attribute independently without considering the dependency relationship among attributes, we treat  $X$  and  $Y$  as if they are one attribute.

We assume that the adversaries have the disguised data  $Y = X + R$ , where  $X$  is the original data, and  $R$  is the noise with a zero mean. Let  $X$  have  $n$  records or objects, which are considered as realizations of  $n$  independent identically distributed (i.i.d.) random variables or i.i.d. random vectors (when there are multiple attributes). Let  $R$  have the same size of data values as  $X$ . They are the realizations of  $n$  independent random variables or random vectors, drawn from a certain distribution.

#### 4.1 Using Noise Distribution

This is a naive guessing method: for each disguised data item  $y$ , the adversaries always use  $y$  as its guess of the original, i.e., the adversaries always guess the value of the random noise to be zero. We call this method the *Noise Distribution-based Reconstruction (NDR)*.

Let  $y_i = x_i + r_i$  for  $i = 1, \dots, n$ , where  $x_i$ ,  $y_i$ ,  $r_i$  are samples of  $X$ ,  $Y$ , and  $R$ , respectively. The mean square error (m.s.e.) of the NDR scheme can be derived in the following:

$$m.s.e. = \frac{1}{n} \sum_{i=1}^n (y_i - x_i)^2 = \frac{1}{n} \sum_{i=1}^n r_i^2 = \frac{1}{n} \sum_{i=1}^n (r_i - 0)^2$$

From the above equation, the m.s.e. of NDR is exactly the variance of the random numbers. When the random numbers have a large variance, the reconstruction accuracy of NDR is low.

#### 4.2 Using Univariate Distribution

NDR scheme is not good for reconstructing the original data. It does not consider the distribution of  $X$  and  $Y$ , which can be helpful for data reconstruction.

In this subsection, we show how to reconstruct the original data for each attribute based on some distributions. Since we treat each attribute of the data set independently, we call this method the *Univariate Distribution-based Reconstruction (UDR)*.

Let  $f_X$ ,  $f_Y$ ,  $f_R$  represent the distribution of  $X$ ,  $Y$ , and  $R$ , respectively. We first derive the posterior distribution  $P(X|Y)$ , which gives us the probability for different values of  $X$  after having observed the value of  $Y$ . Since our goal is to reconstruct the original data, we need to pick a value that can minimize the overall mean square error. Our next theorem indicates that picking the expected value of the distribution achieves the minimum mean square error:

**THEOREM 4.1.** *Given a distribution  $f(x)$ , let  $\bar{x}$  be the expected value of  $x$ . Let  $z$  be a constant. The mean square error  $e = \int_{-\infty}^{\infty} (x - z)^2 f(x) dx$  is minimized when  $z = \bar{x}$ .*

**PROOF.** If we want to find what value of  $z$  makes the  $e = \int_{-\infty}^{\infty} (x - z)^2 f(x) dx$  minimum, we can differentiate the equation twice on  $z$ . Then we find a value which makes the first derivative equal to zero and the second derivative larger than zero. This value indeed minimizes the value of  $e$ . The

first derivative is:

$$\frac{\partial \int_{-\infty}^{\infty} (x-z)^2 f(x) dx}{\partial z} = 0$$

then

$$\begin{aligned} \int_{-\infty}^{\infty} 2 * (x-z) f(x) dx &= 0. \\ 2 * \int_{-\infty}^{\infty} x f(x) dx - 2z * \int_{-\infty}^{\infty} f(x) dx &= 0 \\ \int_{-\infty}^{\infty} x f(x) dx - z &= 0 \\ z &= \bar{x} \end{aligned} \quad (1)$$

The second derivative is:

$$\frac{\partial^2 \int_{-\infty}^{\infty} (x-z)^2 f(x) dx}{\partial^2 z} = 2 > 0 \quad (2)$$

Therefore to minimize mean square errors,  $z$  must be the expected value of  $x$ .  $\square$

Next we will show how to compute the posterior distribution  $P(X|Y)$  and its expected value. To compute  $P(X|Y)$ , we need to know the distributions  $f_X$ ,  $f_Y$ , and  $f_R$ .  $R$ 's distribution  $f_R$  is public.  $Y$ 's distribution  $f_Y$  can be estimated from the samples, i.e., the disguised data set.  $X$ 's distribution  $f_X$  is unknown; however, it has been shown by the studies in the privacy preserving data mining area that  $f_X$  can be estimated from the disguised data [2]. Therefore, in our next analysis, we assume all three distributions  $f_X$ ,  $f_Y$ , and  $f_R$  are known. We have the following:

$$P(x|Y = y) = \frac{f(y|x)f_X(x)}{f_Y(y)} = \frac{f_R(y-x)f_X(x)}{f_Y(y)}. \quad (3)$$

Therefore the expected value of  $X$  given the disguised value  $Y = y$  is the following:

$$\begin{aligned} E(x|Y = y) &= \int_{-\infty}^{\infty} x \frac{f_X(x)f_R(y-x)}{f_Y(y)} dx \\ &= \frac{\int_{-\infty}^{\infty} x f_X(x)f_R(y-x) dx}{f_Y(y)}. \end{aligned} \quad (4)$$

We thus use  $E(x|Y = y)$  as our guess to reconstruct the original data.

It should be noted that UDR only considers the distribution of one dimension; it does not use any correlation between different attributes. If the attributes are highly correlated, the use of the correlations will greatly help the adversaries' estimations. In the following sections, we will study how to take advantage of the correlations among the attributes.

## 5. PCA-BASED DATA RECONSTRUCTION

In this section, we will present a different estimation technique which is based on PCA (principal component analysis). We called this technique PCA-Based Data Reconstruction (PCA-DR). To help readers understand PCA-DR, we briefly describe how PCA works.

### 5.1 Principal Components Analysis

Principal Component Analysis (PCA)[14] is a way to reduce the dimensionality of a data set with interrelated variables, but still contain as much variance of the data set as possible. If a data set has  $m$  variables, each of which has  $n$  implementations, PCA can transform the data set to a new data set with  $p \leq m$  variables, which are uncorrelated and are ordered by the variances they contain. We usually say there is a strong trend along a direction if the variance in the direction is large.

Let  $D$  be a data set of  $n$  records of  $m$  variables (also called attributes). It can be viewed as a transposed vector of  $m$  variables. Let us start to search for the first principal component (PC), which presents the largest variance of the original data set in the direction of a certain vector. We look for a linear function  $De_1$  of the variables of  $D$  which has maximum variance, where  $e_1$  is a vector of  $m$  constants. To get the second PC, we look for a linear function  $De_2$ , uncorrelated with  $De_1$ , and having maximum variance. Accordingly, a linear function  $De_p$ , uncorrelated with  $De_1, \dots, De_{p-1}$ , is found which has maximum variance. The result vectors  $De_1, De_2, \dots, De_p$  are called principal components (PCs). Since the value of  $p$  is always smaller than or equal to  $m$ , PCA is used for compression. The variances in the directions of the vectors decrease from  $De_1$  to  $De_p$ .

If some variables have significant correlations among them, the first few generated PCs will count most of the variances in  $D$ . Accordingly, the subsequently-generated PCs will count a smaller portion of the variances of  $D$ .

In order to find PCs, the covariance matrix  $C$  is computed. This is the matrix whose  $(i, j)$ -th entry is the covariance between the  $i$ th and  $j$ th variables of  $D$  (when  $i = j$ , it is the variance of the  $i$ th attribute of  $D$ ). Then,  $e_k$  is an eigenvector of  $C$  corresponding to its  $k$ th largest eigenvalue  $\lambda_k$ . The  $k$ th PC is  $De_k$ , the variance of which is equal to  $\lambda_k$ . We briefly introduce the procedures of decreasing the number of the data dimensions and of restoring data below.

#### 5.1.1 Decreasing the Dimensionality

Let the original data set be  $D$  as before. The mean of each attribute of the data set is 0 due to the requirement of PCA. A non-0-mean data set can always be adjusted to a 0-mean data set by subtracting the mean of each attribute from it. Then all operations can be executed on the adjusted data set. When the operations are done and restoring the data set is wanted, the mean will be added back. For the simplicity of presentation, we ignore the adjustment steps and consider all data sets we use here are 0-mean data sets.

From  $D$ , its covariance matrix  $C$  can be computed. Using  $C$ , eigenvectors  $[e_1, e_2, \dots, e_m]$  can then be obtained, so is their corresponding eigenvalues:  $\lambda_1, \dots, \lambda_m$ , where  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_m$ .

Assume that the data have large variances along the directions of the first  $p$  eigenvectors, and small variances along the directions of the other  $m - p$  eigenvectors. Let  $E = [e_1, e_2, \dots, e_p]$  be a matrix of size  $m * p$ . The following equation describes the transformation that can reduce dimensions.

$$D_n = DE, \quad (5)$$

where  $D_n$  is a new data matrix of size  $n * p$ . Thus, the number of the dimensions of the data set decreases from  $m$

to  $p$ .

### 5.1.2 Restoring the Original Data

Next we restore the original data from the new data. If  $p = m$ ,  $E$  is orthogonal, meaning that its transpose is its inverse. We have:

$$D = D_n(E)^{-1} = D_n E^T$$

If  $E$  is only composed of  $p$  ( $< m$ ) eigenvector, the above equation is only an approximation and becomes

$$\hat{D} = D_n E^T \quad (6)$$

where  $\hat{D}$  is the estimated original data set.

When we use reduced  $E$  (e.g. the first  $p$  eigenvectors), the restored data will not reflect the variances along the directions of the other  $m - p$  eigenvectors. However, the restored data still contains most information of the data set because the variances in the directions of the principal eigenvectors are maintained. We call an eigenvector ‘‘principal’’ if it is used to calculate the principal components. We call an eigenvalue ‘‘principal’’ when it is corresponding to a principal eigenvector.

## 5.2 PCA-Based Data Reconstruction

Existing methods have not exploited the correlations when quantifying privacy. We believe that the correlations can help the adversaries make more accurate guess. For example, let  $A_i$  be an attribute in the data set. If several attributes are highly correlated with  $A_i$ , then we have redundant information about  $A_i$ . We should be able to estimate  $A_i$  with better accuracy based on this information.

When a data set has strong correlations among its data, the data set has large variances in the directions of some vectors but small variances in the other directions. The addition of noise does not change the trends too much, because if the random numbers added to the original data are independent, their variances will be evenly distributed among all the directions.

Principal Component Analysis is a technique that identifies those trends. Let us consider the information loss when we only select  $p$  principals out of the total  $m$ . There will be information loss. All the variance along the other  $m - p$  directions will be lost. However, when the data are highly correlated, the variances along the first  $p$  directions are much larger than the variances along the rest  $m - p$  directions. Thus removing those  $m - p$  directions during the transformation does not cause much information loss.

The information loss for the random numbers is different. In randomization scheme, random numbers are independent for each attribute. Their correlations are zero. Therefore, their variance will be evenly distributed to those  $m$  directions. If we remove  $m - p$  directions in the PCA transformation, we are able to remove  $\frac{m-p}{m}$  portion of the random numbers’ variance. The more variance of random numbers we remove, the better.

Therefore, if the data is highly correlated, then more dimensions can be reduced without causing too much information loss for the original data; at the same time, the information loss for the noise increases. Intuitively speaking, during the PCA transformation using the first  $p < m$  principals, we can filter out a portion of the random numbers.

Based on the above observation, we present a PCA-based data reconstruction scheme. Since PCA introduces infor-

mation loss, it is important to understand how much of the original information is lost and how much of the noise is lost. Using the PCA-based scheme proposed by Kargupta et al. [16], the information loss and noise loss cannot be clearly quantified, because of matrix perturbation theory also used in that scheme complicates the analysis. Therefore, to be able to understand how correlation affect the privacy of the randomization scheme, we choose to use PCA directly to reconstruct the data. Namely we reconstruct the original eigenvalues and eigenvectors, and then based on the original eigenvalues, we select the principal components. Since the original eigenvalues reflect the degree of correlations among data attributes, the number of principal components and the sum of their variances indicate how much of the original information is kept via PCA.

### 5.2.1 Estimating Covariance Matrix

To conduct PCA, we need to know the covariance matrix for the original data. How do we get the covariance matrix for the original data? The following theorem provides the answer.

**THEOREM 5.1.** *Let  $X_i$  and  $X_j$  represent two variables in the data set.  $Cov(X_i + R_i, X_j + R_j)$  represents the  $(i, j)$ -th entry of the covariance matrix from the disguised data set, where  $R_i$  and  $R_j$  are random variables with zero means, and they are independent from  $X_i$  or  $X_j$ .  $Cov(X_i, X_j)$  represents the  $(i, j)$ -th entry of the covariance matrix from the original data set. Let  $\sigma$  be the standard deviation of  $R_i$ . We have the following relationship:*

$$\begin{aligned} & Cov(X_i + R_i, X_j + R_j) \\ = & \begin{cases} Cov(X_i, X_j) + \sigma^2, & \text{for } i = j \\ Cov(X_i, X_j), & \text{for } i \neq j \end{cases} \end{aligned}$$

**PROOF.** Based on the definition of the covariance, we have the following equations:

$$\begin{aligned} & Cov(X_i + R_i, X_j + R_j) \\ = & E((X_i + R_i)(X_j + R_j)) - E(X_i + R_i)E(X_j + R_j) \\ = & E(X_i X_j) + E(R_i)E(X_j) + E(X_i)E(R_j) \\ & + E(R_i R_j) - E(X_i)E(X_j) \\ = & E(X_i X_j) + 0 + 0 + E(R_i R_j) - E(X_i)E(X_j) \\ = & Cov(X_i, X_j) + E(R_i R_j). \end{aligned}$$

When  $i \neq j$ ,  $R_i$  and  $R_j$  are independent, so  $E(R_i R_j) = E(R_i)E(R_j) = 0$ ; when  $i = j$ ,  $E(R_i R_i) = E((R_i - 0)^2) = \sigma^2$ , where  $\sigma$  is the standard deviation of the random variables  $R_i$ . Combining this with the above equation, we have proved the theorem.  $\square$

When the sample size is large enough, the relationship presented in the above theorem carries over to the sample covariance matrices (PCA is applied to sample covariance matrices). The above theorem indicates that we can derive the covariance matrix of the original data based on the disguised data. All we need to do is to subtract  $\sigma^2$  from the diagonal elements of the covariance matrix that is derived from the disguised data. Although the derived matrix is only an approximation, when the number of samples becomes larger, the approximation becomes more accurate.

### 5.2.2 Applying PCA

After getting the approximated sample covariance matrix for the real data, we can use the following PCA-based method to reconstruct the original data (recall that  $Y$  represents the disguised data. We use  $C$  to represent the sample covariance matrix derived from  $Y$ ):

1. Conduct PCA to get  $C = Q\Lambda Q^T$ , where  $\Lambda$  is a diagonal matrix consisting of eigenvalues, and  $Q$  a matrix formed by eigenvectors.
2. Let  $p$  be the number of principal components to be selected. Set  $\hat{Q}$  to be the first  $p$  columns of  $Q$ .<sup>1</sup>
3. Reconstruct the data using  $\hat{X} = Y\hat{Q}\hat{Q}^T$ .

Let  $m$  be the number of attributes in the data set. If  $p = m$  (i.e., we do not reduce the dimension), the above reconstruction procedure gets back to  $Y$ , and nothing is filtered out. When  $p < m$ ,  $\hat{X}$  will lose information. If we lose the same amount of information on both  $X$  and  $R$ , then such a transformation is not helpful. However, as we have discussed earlier, when the data are highly correlated, we do not lose much information on  $X$ ; more importantly, we lose information about  $R$ , and the amount of information loss with regard to  $R$  should, intuitively speaking, be proportional to the ratio of  $\frac{p}{m}$ . In the next sub-section, we will formally quantify the information loss on  $R$ .

### 5.3 Analysis

For the sake of simplicity, we only analyze PCA-DR using covariance matrix from the original data. That is, the covariance matrix is directly obtained from the original data, rather than being estimated from the disguised data. There are only minor differences between the covariance matrices from original data and the estimated one. From the last step of PCA procedure described earlier, we have

$$\hat{X} = Y\hat{Q}\hat{Q}^T,$$

so we get

$$\begin{aligned} \hat{X} &= (X + R)\hat{Q}\hat{Q}^T \\ &= X\hat{Q}\hat{Q}^T + R\hat{Q}\hat{Q}^T. \end{aligned} \quad (7)$$

The error between  $\hat{X}$  and the original data  $X$  comes from two sources: one is the error caused by  $X\hat{Q}\hat{Q}^T$ , the other is the error caused by  $R\hat{Q}\hat{Q}^T$ . The former is decided by the correlations among data and the number of principals included in  $\hat{Q}$ . The latter can be quantified by the following theorem:

**THEOREM 5.2.** *Let  $m$  be the number of the attributes of the original data set,  $p$  be the number of principal components being used in PCA-DR. The variance of the random noise is  $\sigma^2$ , and its mean is 0. Let  $\delta^2$  be the mean square error of PCA-DR caused by  $R\hat{Q}\hat{Q}^T$ . We have the following relationship:*

<sup>1</sup>There are a number of ways to select principal components. We can fix the number of selected principal components; we can also fix the portion of the original information that we want to keep; we can also choose the dominant eigenvalues by finding the largest gap between the dominant eigenvalues and the non-dominant ones. The last method is used in our experiments.

$$\delta^2 = \sigma^2 \frac{p}{m}. \quad (8)$$

PROOF. See Appendix A.  $\square$

Due to Theorem (5.2), the mean square error that is caused by  $R$  in PCA-DR scheme is proportional to the variance of the random numbers and the ratio of  $\frac{p}{m}$ . This confirms our intuitive explanation. We will also use experiments to verify these relationships.

## 6. BAYES-ESTIMATE-BASED DATA RECONSTRUCTION

The PCA-based reconstruction works well when the data are highly correlated. However, when the correlations of data are not high enough, the non-principal components will not be many; according to Theorem (5.2), we will not be able to filter out significant amount of the noise. Our results in Section 7 will show that when the correlations become low, the accuracy achieved by the PCA-based scheme is even worse than the univariate data reconstruction scheme.

In this section, we describe a more accurate data reconstruction method. We want to fully take advantage of the relationship among data. Unlike the PCA-based schemes, which use this relationship to filter out noise, We formulate the data reconstruction problem as an estimation problem, i.e., based on the disguised data that we have observed and on the data relationship that we know, we use a value that can most likely produce such an observation as our reconstructed data. In other words, given the disguised data  $Y$ , we search for  $X$ , such that the posterior distribution  $P(X | Y)$  is maximized. We then use  $X$  as the data reconstructed from the disguised data. This is the idea of the Bayes estimate [20]. We call our scheme the *Bayes-Estimate-based Data Reconstruction (BE-DR)*.

To simplify derivation, we assume that the original data have multivariate normal distribution. This assumption is reasonable since this distribution is a good approximate distribution in many situations [13]. Because of the appealing properties of multivariate normal distribution, the calculation of Bayes estimate is simplified to computation of matrices and vectors even though the form of the distribution is more complicated. As we will explain later, this assumption can be relaxed.

### 6.1 Data Reconstruction

Suppose that random noise used for each attribute has normal distribution and it is independent from those used for other attributes. Suppose that an adversary only has the disguised data set and the distribution of the random noise. Let the original data have  $m$  attributes and  $n$  records. They can be considered as observations of a random vector of length  $m$ . Let the random vector of the original data be  $\vec{X}$ . Similarly let the random vector of the noise be  $\vec{R}$ , and the random vector of the disguised data  $\vec{Y}$ . Let  $f_{\vec{X}}$ ,  $f_{\vec{Y}}$ ,  $f_{\vec{R}}$  represent the distributions of  $\vec{X}$ ,  $\vec{Y}$ ,  $\vec{R}$ , respectively. The distribution of  $\vec{X}$  and  $\vec{R}$  are described in the following:

$$\begin{aligned} f_{\vec{X}}(\vec{x}) &= \frac{1}{(2\pi)^{m/2} |\Sigma_x|^{1/2}} e^{-\frac{1}{2}(\vec{x} - \mu_x)^T \Sigma_x^{-1} (\vec{x} - \mu_x)} \\ f_{\vec{R}}(\vec{r}) &= \frac{1}{(2\pi\sigma)^{m/2}} e^{-\frac{1}{2}(\vec{r} - \mu_r)^T (\vec{r} - \mu_r) / \sigma^2}, \end{aligned}$$

where  $\Sigma_x$  is the covariance matrix of the original data,  $\sigma^2$  is the variance of the random noise,  $\vec{\mu}_x$ ,  $\vec{\mu}_r$  are the mean vectors of the original data and the noise data.

We use the posterior distribution  $P(\vec{X}|\vec{Y})$  to represent the probabilities for different values of  $\vec{X}$  given an observation of  $\vec{Y}$ . By using Bayesian rule, we get the following formulae:

$$P_{\vec{X},\vec{Y}}(\vec{x}|\vec{y}) = \frac{f_{\vec{X}}(\vec{x})f_{\vec{Y}|\vec{X}}(\vec{y}|\vec{x})}{f_{\vec{Y}}(\vec{y})}, \quad (9)$$

where  $f_{\vec{Y}|\vec{X}}(\vec{y}|\vec{x})$  represents the probability of getting  $\vec{y}$  from  $\vec{x}$ , which is exactly the probability of the random number  $(\vec{y} - \vec{x})$ . Therefore,  $f_{\vec{Y}|\vec{X}}(\vec{y}|\vec{x}) = f_{\vec{R}}(\vec{y} - \vec{x})$ .

We want to find a value of  $\vec{X}$ , such that  $P_{\vec{X},\vec{Y}}(\vec{x}|\vec{Y} = \vec{y})$  is maximized. Noticing that the denominator  $f_{\vec{Y}}(\vec{y})$  does not change when  $\vec{X}$  changes, we only need to consider the numerator. Thus we only need to maximize:

$$\begin{aligned} f_{\vec{X}}(\vec{x})f_{\vec{Y}|\vec{X}}(\vec{y}|\vec{x}) &= f_{\vec{X}}(\vec{x})f_{\vec{R}}(\vec{y} - \vec{x}) \\ &= \frac{1}{(2\pi)^{m/2}|\Sigma_x|^{1/2}} e^{-\frac{1}{2}(\vec{x} - \vec{\mu}_x)^T \Sigma_x^{-1} (\vec{x} - \vec{\mu}_x)} \cdot \\ &\quad \frac{1}{(2\pi\sigma)^{m/2}} e^{-\frac{1}{2}(\vec{y} - \vec{x} - \vec{\mu}_r)^T (\vec{y} - \vec{x} - \vec{\mu}_r) / \sigma^2} \end{aligned} \quad (10)$$

Since the logarithm is a monotone one-to-one function, we could maximize the following function instead:

$$-\frac{1}{2}(\vec{x} - \vec{\mu}_x)^T \Sigma_x^{-1} (\vec{x} - \vec{\mu}_x) - \frac{1}{2}(\vec{y} - \vec{x})^T (\vec{y} - \vec{x}) / \sigma^2.$$

Note that in the above equation the constant terms are ignored because it does not affect computing the maximum estimator of  $\vec{x}$ ; the mean vector  $\vec{\mu}_r$  is ignored too since it is assumed to be zero vector in randomization schemes.

We let the first derivative of the above equation with respect to  $\vec{x}$  be 0. We get:

$$\Sigma_x^{-1}(\vec{x} - \vec{\mu}_x) + (\vec{x} - \vec{y}) / \sigma^2 = 0.$$

After simplifying and rearranging the above equation, we have

$$\hat{\vec{x}} = (\Sigma_x^{-1} + 1/\sigma^2 \cdot I)^{-1} (\Sigma_x^{-1} \vec{\mu}_x + \vec{y} / \sigma^2), \quad (11)$$

where  $I$  is the identity matrix of the same size as  $\Sigma_x$ .

Equipped with Equation (11), we now describe our Bayes-Estimate-based data reconstruction scheme:

1. Derive  $\Sigma_x$  from Theorem (5.1) using disguised data  $Y$ .
2. Derive  $\vec{\mu}_x$  by computing the mean vector of the disguised data. We know that  $\vec{\mu}_x \approx \vec{\mu}_y$  because random noises have zero means.
3. For each  $\vec{y}$ , derive  $\hat{\vec{x}}$  using Equation (11).
4. Use  $\hat{\vec{x}}$  as the reconstructed value.

The BE-based scheme, the PCA-based scheme, and the univariate data reconstruction scheme have the following relationship: when the correlations among data are low, e.g., data are independent, the results of BE-DR should converge to the univariate data reconstruction. This is because when data are independent, data from one attribute cannot help the reconstruction of another attribute. Thus, the BE-DR

scheme is equivalent to the univariate data reconstruction. On the other hand, when the correlations among data become high, the results of BE-DR should be similar to those of PCA-DR, because they both fully exploit the correlation-ship among data.

Regarding our assumption on the multivariate normal distribution: although we have only shown the results for data sets that satisfy multi-normal distribution, the approach can be extended to data sets that satisfy other distribution. However, for other distributions, we might not be able to derive an equation with a simple analytic form for its first derivative. In such situations, the Bayes estimate must be sought using numerical methods, such as Gradient descent methods [12, 3]. We will study them in our future work.

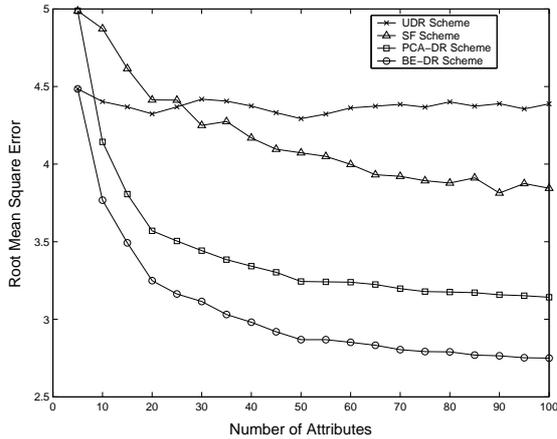
## 7. EXPERIMENT

### 7.1 Methodology

We have designed a series of experiments to evaluate the PCA-DR scheme and the BE-DR scheme. Our goal is to find out how the correlations among the attributes affect the accuracy of these methods. Data correlations can be affected by a number of parameters, including the ratio of the number of principal components to the number of attributes and the variance of data on the principal and non-principal components. We have designed experiments to study how these parameters affect our schemes. We also compare our results with SF algorithm [16].

We decided to use synthetic data for our experiments, because it is difficult to find real data sets that bear properties pre-determined for each experiment. Our approach is to generate a covariance matrix first, then generate the synthetic data set based on the covariance matrix, and finally conduct the PCA-DR or BE-DR procedure. However, generating a covariance matrix with pre-determined properties is not a trivial task either. To better control the properties of the matrix, we generated the covariance matrix in a reverse way: we generated the eigenvalues and eigenvectors first, and then we computed the covariance matrix from the eigenvalues and eigenvectors. We can control the properties of covariance matrix by changing eigenvalues. Our procedure is described in the following:

1. We specify  $\Lambda$  as a diagonal matrix with the corresponding eigenvalues on its diagonal. The size of  $\Lambda$  is  $m$  by  $m$ .
2. By using Gram-Schmidt orthonormalization process [4], we generate an orthogonal matrix  $Q$  of size  $m$  by  $m$ . We consider each column of  $Q$  as an eigenvector.
3. We let the covariance matrix  $C = Q \times \Lambda \times Q^T$ .
4. We generate a data set based on the covariance matrix. In our experiments, we use `mvrnd` from Matlab to generate data from  $C$ . The function `mvrnd` generates a data set of multivariate normal distribution based on the provided covariance matrix and the mean vector. This resultant data set will be used as the original data set  $X$ .
5. We randomly generate a noise data set using normal distributions. This noise data set is then added to the original data set to form the disguised data set  $Y$ .



**Figure 1: Experiment 1: Increase the Number of Attributes**

**Benchmark.** The objective of this paper is to show how much the correlations among the attributes can help disclose private information. Therefore, we compare our results with the data reconstruction method that does not consider correlations. UDR (Univariate Data Reconstruction) described in Section 4 is such a data reconstruction scheme. We use its results as our baseline comparison. The difference between our PCA-DR, BE-DR results and the UDR results indicates how much the correlations can help disclose private information.

## 7.2 Experiment 1: Increasing the Size of the Covariance Matrix

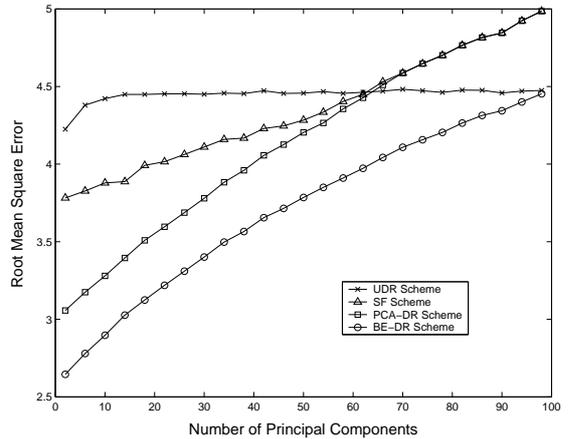
In this experiment, we change the correlations among data by increasing the number of attributes while fixing the number of principal components. We first fixed the number of principal components to  $p$  by letting the first  $p$  eigenvalues in  $\Lambda$  to be  $\lambda$ , and the other  $m - p$  eigenvalues to be relatively small numbers.  $p$  is 5 in the experiment. We then increased the size of  $\Lambda$  from  $p$  to  $m$ .

Since UDR is used as the benchmark, we would like to keep its result a constant when we generate different data sets for this experiment. However, we found that this was not an easy task. Since the mean square error of UDR only depends on the standard deviation of the original data (we have already fixed the distribution of the disguising noises), to keep it the same, we should keep the average standard deviation of each attribute the same. Due to the following property of eigenvalues, by selecting the eigenvalues that satisfy this property, we can keep the results of UDR almost the same:

$$\sum_{i=1}^m \lambda_i = \sum_{i=1}^m a_{ii}, \quad (12)$$

where  $a_{ii}$  is the diagonal element of the covariance matrix, i.e., it is the variance of the  $i$ th attribute.

Once the data is generated, we use the UDR, PCA-DR, BE-DR and SF schemes to reconstruct the original data from the disguised data, and measure the mean square errors between the reconstructed data and the original data. The results are depicted in Figure 1.



**Figure 2: Experiment 2: Increase the Number of Principal Components**

The experimental results clearly show that all the correlation-based reconstruction schemes (SF, PCA-DR, and BE-DR) have lower reconstruction errors when the number of attributes increase. Since the number of principal components in this experiment is fixed, the larger the number of the attributes, the higher the correlations, and as this experiment indicates, the more accurately one can reconstruct the original data. We also see that UDR scheme is not sensitive to the change of correlations because it does not exploit such a relationship. The performance of UDR is much worse compared to the other schemes when the data correlations are high.

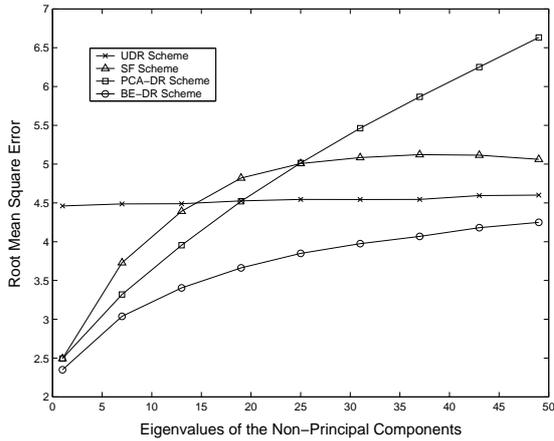
We also observed that SF scheme does not perform as well as the PCA-DR scheme in this experiment. However, this comparison result does not always hold. The key difference of the SF scheme and the PCA-DR scheme is how to separate the principal components and the non-principal components. SF scheme uses the bounds derived from the matrix perturbation theory to identify and separate the principal components from the non-principal components. Our experiments show that when the eigenvalues on the non-principal components are not very small, the derived bounds tend not to be quite accurate. That might be the cause of SF's worse performance. In our experiment 3, we will show that when the eigenvalues on the non-principal components are small, the performance of SF and that of PCA-DR are indeed close.

Most importantly, the experiment shows that BE-DR achieves better performance than PCA-DR and SF schemes. This result is consistent throughout all our experiments. The reason is that the BE-DR method utilizes all the information, including the non-principal components, while the PCA-based schemes discard the non-principal components.

## 7.3 Experiment 2: Increasing the number of principal components

In this experiment, we change the correlations among data from another angle: instead of changing the value of  $m$  as in the previous experiment, we change the value of  $p$ , the number of principal components. When  $p$  increases, the number of principal components increases, thus the correlations among data decrease.

We fixed the size of the covariance matrix to be  $100 * 100$ ,



**Figure 3: Experiment 3: Increase the Eigenvalues of the non-Principal Components**

i.e., there were 100 attributes. We let the first  $p$  eigenvalues in matrix  $\Lambda$  be  $\lambda$ , and the rest  $m - p$  eigenvalues be very small numbers compared with  $\lambda$ . This way, the number of principal components is  $p$ . In the experiment, we increase the value of  $p$  from 2 to 100. We use the same method as the previous experiment to maintain the result of UDR method a constant.

The results of this experiment are depicted in Figure 2. The results show that SF, PCA-DR and BE-DR achieve better accuracy when the number of principal components becomes less, i.e., the correlations become higher.

For PCA-based schemes, intuitively speaking, when the number of principal components increases, the correlations among the attributes become weaker. Thus we need to keep more directions in the PCA-based data compression. As a result, more noises are kept because they are evenly distributed along all directions; therefore the accuracy of the data reconstruction decreases.

For the same reason as the previous experiment, the BE-DR scheme demonstrates a better performance than the PCA-DR schemes.

### 7.4 Experiment 3: Increasing the Eigenvalues of non-principal Components

Eigenvalues of non-principal components represent the discarded variances on those non-principal directions. They can affect the reconstruction results of PCA-based schemes and BE-based scheme. In this experiment, we examine how the eigenvalues influence the estimation results of PCA-DR and BE-DR.

We fixed the size of  $\Lambda$  to  $100 * 100$ . We let the first 20 eigenvalues be  $\lambda$  and the other 80 eigenvalues be variables. Here we let them change from 1 to 50, which is still smaller than  $\lambda (=400$  in our experiment).

The results are depicted in Figure 3. From the results, changing the eigenvalues of the non-principal components does not affect UDR, but it does significantly affect SF, PCA-DR and BE-DR. When the eigenvalues become larger or the correlations of the original data are low, the accuracy of SF, PCA-DR and BE-DR all become worse. Because SF and PCA-DR discard the information along the non-principal eigenvectors, when the eigenvalues of those eigen-

vectors become larger, more information of the original data is discarded; thus the estimation errors become higher. After certain points, the original information is discarded so much that the errors of SF and PCA-DR schemes are even higher than UDR.

BE-DR scheme does not have the above drawbacks. As demonstrated by the figure, the performance of BE-DR converges to the performance of UDR when the correlations of data are low. Note that UDR also “implicitly” uses the Bayes estimate principle, but instead on single attribute. When data correlations are low, the data become more and more independent; thus multivariate Bayes estimate reconstruction becomes equivalent to univariate data reconstruction because no correlation among attributes can be exploited.

## 8. IMPROVING PRIVACY PRESERVATION OF DATA RANDOMIZATION

### 8.1 An Improved Randomization Scheme

From the analysis of PCA, we know that the variances of the random noises are evenly distributed among both principal and non-principal components, while most of the information of the original data concentrates on the principal components. Therefore, when we discard those non-principal components, we can remove (or filter) far more noises than what we do to the original data. However, if random noises also concentrate on the principal components, separating original data from random noises becomes difficult.

Motivated by the above observation, we propose to use correlated random noises to disguise original data. In particular, we let the correlations of the random noises similar to the correlations of the original data. For example, when the original data have a multivariate normal distribution with covariance matrix  $C$ , we generate the random noises using the same covariance matrix. This guarantees that noises also concentrate on the principal components (of the original data).

Under the new randomization scheme, the PCA-based data reconstruction scheme is still the same, but the formula in Equation (11) for BE-based scheme needs to be modified. We have the following formula for BE-DR.

**THEOREM 8.1.** *Let the covariance matrix of the original data be  $\Sigma_x$ , and the covariance matrix of the random noises be  $\Sigma_r$ . Let the mean vectors of original data and noise data be  $\vec{\mu}_x$  and  $\vec{\mu}_r$ , respectively. Let the disguised data vector  $\vec{y}$ . Then the Bayes estimate of the original data vector is*

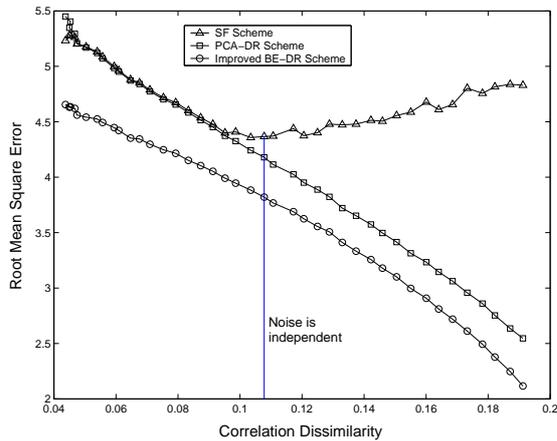
$$\hat{\vec{x}} = (\Sigma_x^{-1} + \Sigma_r^{-1})^{-1}(\Sigma_x^{-1}\vec{\mu}_x - \Sigma_r^{-1}\vec{\mu}_r + \Sigma_r^{-1}\vec{y}). \quad (13)$$

**PROOF.** Similar to the derivation of Equation (11)  $\square$

Because the original data cannot be directly observed, we do not have the covariance matrix  $\Sigma_x$ . However, we can estimate  $\Sigma_x$  using the disguised data  $Y$  and the covariance matrix of the random noise. The following theorem describes how to derive  $\Sigma_x$ :

**THEOREM 8.2.** *Let the covariance matrix of the original data, random noise, and the disguised data be  $\Sigma_x$ ,  $\Sigma_r$ , and  $\Sigma_y$ , respectively. We have the following relationship:*

$$\Sigma_y = \Sigma_x + \Sigma_r. \quad (14)$$



**Figure 4: Experiment 4: Increasing the correlation dissimilarity of the original data and random noise.**

PROOF. Similar to the proof of Theorem 5.1  $\square$

There are two important aspects of privacy preserving data mining. One is to preserve the original data’s privacy, the other is to do the data mining. Therefore, regardless of how the random noises are generated, the disguised data set should be useful for data mining. Equation (14) indicates that when the data are multivariate normal, we can reconstruct the probability distribution of the original data from the disguised data set. This distribution is sufficient for many data mining computations. Actually, the reason why randomization can be used in achieving privacy-preserving data mining is that those supported data mining operations are based on aggregate information (e.g. distribution). Therefore, our improved randomization scheme can still be used in many privacy-preserving data mining computations.

## 8.2 Experiment Results

To understand how the correlations among random noises affect the data reconstruction methods, we use random noises with different levels of “similarity” compared to the covariance matrix  $\Sigma_x$  of the original data. We hypothesize that the privacy is better protected when the noises are more “similar” to  $\Sigma_x$ . We use the following *correlation dissimilarity* metric to quantify the difference between the correlations among random noises and the correlations among the original data.

**DEFINITION 8.1. (Correlation Dissimilarity)** Let two data sets be  $X, R$  with the same size of  $n$  by  $m$ . Let the matrix of correlation coefficients for  $X$  be  $C_X$ . Let the matrix of correlation coefficients for  $R$  be  $C_R$ . The correlation dissimilarity between  $X$  and  $R$  is defined as:

$$Dis(X, R) = \frac{1}{m^2 - m} \sqrt{\sum_{i=1}^m \sum_{j=1, j \neq i}^m (C_X(i, j) - C_R(i, j))^2},$$

where  $(i, j)$  means an element in the  $i$ th row and  $j$ th column. Note that, we do not consider the diagonal elements of the matrices of correlation coefficients because they are always 1 and should not be counted for the dissimilarity.

Similar to the experiments in Section 7, we generate an original data set with 100 attributes, where the first 50 eigenvalues have large numbers and the others have small numbers. To make it convenient to control the degree of correlation dissimilarity between the noises and the original data, we generate random noises based on these eigenvalues and the corresponding eigenvectors of the original data. More specifically, we fix the eigenvectors of the noises to be the same as those of the original data, and we then change the values of the eigenvalues. As a result, we will get a new covariance matrix, based on which we can generate random noises. It should be noted that there are many other ways to generate random noises with different correlation dissimilarities.

Our experiment results are depicted in Figure 4. The figure indicates that when the correlations of the random noises are almost the same as that of the original data, data reconstruction has the highest error, i.e., the privacy is best preserved. When the dissimilarity increases, the privacy preservation becomes worse (with the SF algorithm being the exception). The vertical line we draw in the middle of the figure represents the situation where the random noises are not correlated (i.e., the random noises of different attributes are independent). Therefore, the curves on the left side of the vertical line show how much privacy we can gain by using correlated random noises as opposed to independent noises.

On the right hand of the vertical line, noises are still correlated, but they are correlated very differently compared to the original data  $X$ . Actually, based on the ways we generate those noises, the noises on the right side of the vertical line start to concentrate on the non-principal components, whereas the noises on the left side of the line concentrate on the principal components. Our results have shown that the accuracy of PCA-DR and BE-DR continues to increase. This is because the amount of noises used to disguised the principal components of the original data becomes less.

We noticed that the SF algorithm behaves differently after the point of the vertical line. The main reason for this is that the filtering bounds of the SF are derived from independent random noises. It may not optimally filter the noises when noises are correlated.

## 9. CONCLUSION

In this paper, we studied how correlations affect the privacy of a data set disguised via the random perturbation scheme. We presented two methods to reconstruct original data from a disguised data set. One scheme is based on PCA (Principal Component Analysis), and the other scheme is based on the Bayes estimate. Using PCA concepts, we give an intuitive and theoretical explanation on how correlation can affect the data privacy for a randomized data set. Our results have shown that both the PCA-based schemes and the BE-based scheme can reconstruct more accurate data when the correlation of data increases. Our results have also shown that the BE-based scheme is always better than the PCA-based schemes.

To defeat the data reconstruction methods that exploit the data correlation, we proposed a modified random perturbation, in which the random noises are correlated. Our experiments show that the more the correlation of noises resembles that of the original data, the better privacy preservation can be achieved.

Our future work will study how information other than correlation can affect privacy. For example, we will study how partial knowledge of a disguised data set can compromise privacy.

## 10. ACKNOWLEDGMENT

The authors acknowledge supports from the United States National Science Foundation IIS-0219560 and IIS-0312366. The authors would also like to thank all the anonymous reviewers for their valuable comments.

## 11. 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] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. V. der Vorst. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd Edition*. SIAM, Philadelphia, PA, 1994.
- [4] R. Bronson. *Linear Algebra, An Introduction*. Academic Press, 1991.
- [5] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, and M. Y. Zhu. Tools for privacy preserving data mining. *SIGKDD Explorations*, 4(2), December 2002.
- [6] W. Du, Y. S. Han, and S. Chen. Privacy-preserving multivariate statistical analysis: Linear regression and classification. In *Proceedings of the 4th SIAM International Conference on Data Mining*, Lake Buena Vista, Florida, USA, April 2004.
- [7] 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.
- [8] A. Evfimievski, J. E. Gehrke, and R. Srikant. Limiting privacy breaches in privacy preserving data mining. In *Proceedings of the 22nd ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS 2003)*, San Diego, CA, June 2003.
- [9] 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.
- [10] B. Gilburd, A. Schuster, and R. Wolff. A new privacy model and association-rule mining algorithm for large-scale distributed environments. In *10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, Seattle, WA, USA, August 2004.
- [11] S. Goldwasser. Multi-party computations: Past and present. In *Proceedings of the 16th Annual ACM Symposium on Principles of Distributed Computing*, Santa Barbara, CA USA, August 21-24 1997.
- [12] R. Hamming. *Numerical Methods for Scientists and Engineers*. Dover Pubns, 2nd edition, 1987.
- [13] W. Hardle and L. Simar. *Applied Multivariate Statistical Analysis*. Springer-Verlag, 2003.
- [14] I.T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, 1986.
- [15] M. Kantarcioglu, J. Jin, and C. Clifton. When do data mining results violate privacy? In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2004)*, Seattle, WA, USA, August 2004.
- [16] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar. On the privacy preserving properties of random data perturbation techniques. In *the IEEE International Conference on Data Mining*, Melbourne, Florida, November 19 - 22, 2003 2003.
- [17] Y. Lindell and B. Pinkas. Privacy preserving data mining. In *Advances in Cryptology - Crypto2000, Lecture Notes in Computer Science*, volume 1880, 2000.
- [18] D. Meng, K. Sivakumar, and H. Kargupta. Privacy sensitive bayesian network parameter learning. In *The Fourth IEEE International Conference on Data Mining (ICDM)*, Brighton, UK, November 2004.
- [19] B. Pinkas. Cryptographic techniques for privacy-preserving data mining. *SIGKDD Explorations*, 4(2), December 2002.
- [20] H. V. Poor. *An Introduction to Signal Detection and Estimation*. Springer-Verlag, New York, 1994.
- [21] S. Rizvi and J. R. Haritsa. Maintaining data privacy in association rule mining. In *Proceedings of the 28th VLDB Conference*, Hong Kong, China, 2002.
- [22] A. Sanil, A. Karr, X. Lin, and J. Reiter. Privacy preserving regression modelling via distributed computation. In *10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, Seattle, WA, USA, August 2004.
- [23] 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.
- [24] J. Vaidya and C. Clifton. Privacy-preserving k-means clustering over vertically partitioned data. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Ddata Mining*, 2003.
- [25] K. Wang, P. Yu, and S. Chakraborty. Bottom-up generalization: A data mining solution to privacy protection. In *The Fourth IEEE International Conference on Data Mining (ICDM)*, Brighton, UK, November 2004.
- [26] S. L. Warner. Randomized response: A survey technique for eliminating evasive answer bias. *The American Statistical Association*, 60(309):63–69, March 1965.
- [27] R. Wright and Z. Yang. Privacy-preserving bayesian network structure computation on distributed heterogeneous data. In *10th ACM SIGKDD*

## APPENDIX

### A. PROOF OF THEOREM 5.2

Suppose that the random data is a matrix  $R$  of size  $n \times m$ , and the eigenvectors of the covariance matrix of the original data set is  $Q$ :

$$R = \begin{pmatrix} r_{11} & \dots & r_{1m} \\ \vdots & \ddots & \vdots \\ r_{n1} & \dots & r_{nm} \end{pmatrix}, Q = \begin{pmatrix} e_{11} & \dots & e_{1m} \\ \vdots & \ddots & \vdots \\ e_{m1} & \dots & e_{mm} \end{pmatrix}$$

where  $n$  is the number of records and  $m$  is the number of attributes. Let the first  $p$  eigenvectors of  $Q$  be principal and form another matrix  $\hat{Q}$ . Before calculating  $R\hat{Q}\hat{Q}^T$ , we compute the mean square of  $RC$ , where  $C$  is an  $m \times m$  matrix:

$$C = \begin{pmatrix} c_{11} & \dots & c_{1m} \\ \vdots & \ddots & \vdots \\ c_{m1} & \dots & c_{mm} \end{pmatrix},$$

then  $RC$  is:

$$RC = \begin{pmatrix} \sum_{i=1}^m r_{1i}c_{i1} & \dots & \sum_{i=1}^m r_{1i}c_{im} \\ \vdots & \ddots & \vdots \\ \sum_{i=1}^m r_{ni}c_{i1} & \dots & \sum_{i=1}^m r_{ni}c_{im} \end{pmatrix}.$$

The mean square of the first column of  $RC$  is:

$$\frac{1}{n} \sum_{j=1}^n \left( \sum_{i=1}^m r_{ji}c_{i1} \right)^2 = \frac{1}{n} \sum_{i=1}^m \sum_{k=1}^m \left( \sum_{j=1}^n r_{ji}r_{jk}c_{i1}c_{k1} \right)$$

Then we investigate the term in the parenthesis. If  $i = k$ , the term is:

$$\left( \frac{1}{n} \sum_{j=1}^n r_{jt}r_{jt} \right) c_{t1}c_{t1}$$

where  $t$  is any integer from 1 to  $m$ . If  $n$  is large enough, the term becomes:

$$(\sigma^2)c_{t1}c_{t1}.$$

If  $i \neq j$ , the term is:

$$\left( \frac{1}{n} \sum_{j=1}^n r_{js}r_{jt} \right) c_{s1}c_{t1},$$

where,  $s, t$  are any integer from 1 to  $m$ . If considering the  $s$ th and  $t$ th column of the random number to be the variables  $R_s$  and  $R_t$ ,  $E(R_sR_t) = E(R_s)E(R_t) = 0$ . When  $n$  is large enough,  $\frac{1}{n} \sum_{j=1}^n r_{js}r_{jt} = 0$ . Therefore the above term when  $s \neq t$  is 0 when  $n$  is large enough.

Then the mean square of the first column is:

$$\sigma^2 \sum_{t=1}^m c_{t1}^2. \quad (15)$$

Similarly, we have the mean square errors of all columns. We have:

$$\sigma^2 \sum_{t=1}^m c_{tk}^2, k = 1, \dots, m. \quad (16)$$

The the mean square of all columns is

$$\frac{\sigma^2}{m} \sum_{k=1}^m \sum_{t=1}^m c_{tk}^2. \quad (17)$$

Then we know that the mean square of  $RC$  is the variance of the random numbers times the summation of the square of all elements of  $C$  divided by  $m$ .

Let us calculate  $\hat{Q}\hat{Q}^T$ .

$$\begin{aligned} \hat{Q}\hat{Q}^T &= \begin{pmatrix} e_{11} & \dots & e_{1p} \\ \vdots & \ddots & \vdots \\ e_{m1} & \dots & e_{mp} \end{pmatrix} \begin{pmatrix} e_{11} & \dots & e_{m1} \\ \vdots & \ddots & \vdots \\ e_{1p} & \dots & e_{mp} \end{pmatrix} \\ &= \begin{pmatrix} \sum_{i=1}^p e_{1i}e_{1i} & \dots & \sum_{i=1}^p e_{1i}e_{mi} \\ \vdots & \ddots & \vdots \\ \sum_{i=1}^p e_{mi}e_{1i} & \dots & \sum_{i=1}^p e_{mi}e_{mi} \end{pmatrix} \end{aligned} \quad (18)$$

From the previous description, the mean square of  $R\hat{Q}\hat{Q}^T$  is

$$\frac{\sigma^2}{m} \sum_{j=1}^m \sum_{k=1}^m \left( \sum_{i=1}^p e_{ji}e_{ki} \right)^2 = \frac{\sigma^2}{m} \sum_{t=1}^p \sum_{i=1}^p \sum_{j=1}^m \sum_{k=1}^m e_{ji}e_{ki}e_{jt}e_{kt}$$

We investigate the terms in the parenthesis. If  $i = t$ ,

$$\begin{aligned} \sum_{j=1}^m \sum_{k=1}^m e_{ji}e_{ki}e_{jt}e_{kt} &= \sum_{j=1}^m e_{js}^2 \sum_{k=1}^m e_{ks}^2, s = 1, \dots, p \\ &= 1(Q \text{ is orthogonal}). \end{aligned}$$

If  $i \neq t$ ,

$$\begin{aligned} \sum_{j=1}^m \sum_{k=1}^m e_{ji}e_{ki}e_{jt}e_{kt} &= \sum_{j=1}^m e_{ji}e_{jt} \sum_{k=1}^m e_{ki}e_{kt} \\ &= 0(Q \text{ is orthogonal}). \end{aligned}$$

Thus, the mean square of  $R\hat{Q}\hat{Q}^T$  is  $\sigma^2 \frac{p}{m}$ . Then the mean square error of PCA-DR caused by  $R\hat{Q}\hat{Q}^T$  is:

$$\delta^2 = \sigma^2 \frac{p}{m} \quad (19)$$