Random Data Perturbation Techniques and Privacy Preserving Data Mining

Hillol Kargupta and Souptik Datta
Computer Science and Electrical Engineering Department
University of Maryland Baltimore County
Baltimore, Maryland 21250, USA
{hillol, souptik1}@cs.umbc.edu

Qi Wang and Krishnamoorthy Sivakumar
School of Electrical Engineering and Computer Science
Washington State University
Pullman, Washington 99164-2752, USA
{qwang, siva}@eecs.wsu.edu

Extended version of the Best Paper Award winning paper, 2003 IEEE International Conference on Data Mining

Abstract

Privacy is becoming an increasingly important issue in many data mining applications. This has triggered the development of many privacy-preserving data mining techniques. A large fraction of them use randomized data distortion techniques to mask the data for preserving the privacy of sensitive data. This methodology attempts to hide the sensitive data by randomly modifying the data values often using additive noise. This paper questions the utility of the random value distortion technique in privacy preservation. The paper first notes that random objects (particularly random matrices) have "predictable" structures in the spectral domain and then it develops a random matrix-based spectral filtering technique to retrieve original data from the dataset distorted by adding random values. The proposed method works by comparing the spectrum generated from the observed data with that of random matrices. This paper presents the theoretical foundation and extensive experimental results to demonstrate that in many cases random data distortion preserves very little data privacy. The analytical framework presented in this paper also points out several possible avenues for the development of new privacy-preserving data mining techniques. Examples include algorithms that explicitly guard against privacy breaches through linear transformations, exploiting multiplicative and colored noise for preserving privacy in data mining applications.

I. Introduction

Many data mining applications deal with privacy-sensitive data. Financial transactions, health-care records, and network communication traffic are some examples. Data mining in such privacy-sensitive domains is facing growing concerns. Therefore, we need to develop data mining techniques that are sensitive to the privacy issue. This has fostered the development of a class of data mining algorithms [2, 15] that try to extract the data patterns without directly accessing the original data and guarantees that the mining process does not get sufficient information to reconstruct the original data.

This paper considers a class of techniques for privacy-preserving data mining by randomly perturbing the data while preserving the underlying probabilistic properties. It explores the random value perturbation-based approach [2], a well-known technique for masking the data using random noise. This approach tries to preserve data privacy by adding random noise, while making sure that the random noise still preserves the "signal" from the data so that the patterns can still be accurately estimated. This paper questions the privacy-preserving capability of the random value perturbation-based approach. It shows that in many cases, the original data (sometimes called "signal" in this paper) can be closely estimated from the perturbed data using a spectral filter that exploits some theoretical properties of random matrices. It presents the theoretical foundation and provides experimental results to support this claim.

Section II offers an overview of the related literature on privacy preserving data mining. Section III presents the motivation behind the framework presented in this paper. Section IV describes the random data perturbation method proposed in [2]. Section V presents a discussion on the eigenvalues of random matrices that builds the foundation of the technique proposed here to compromise the privacy protection introduced by the random value perturbation-based approach. Section VI presents the intuition behind the theory to separate out random component from a mixture of non-random and random components. Section VII describes the proposed random matrix-based filtering technique to extract the original dataset. Section VIII applies the proposed technique and reports its performance for various data sets. Finally, Section IX concludes this paper and outlines future research directions.

II. Related Work

There exists a growing body of literature on privacy-sensitive data mining. These algorithms can be divided into two different groups. One approach adopts a distributed framework; the other approach adds random noise to the data in such a way that the individual data values are distorted while still preserving the underlying distribution properties at a macroscopic level. The following part of this sections briefly discusses these two approaches.

The distributed approach supports computation of data mining models and extraction of "patterns" at a given node by exchanging only the minimal necessary information among the participating nodes without transmitting the raw data. The field of distributed data mining [16, 26] produced several distributed algorithms that are sensitive to privacy. For example the meta-learning based JAM system [34] was designed for mining multi-party distributed sensitive data such as financial fraud detection. The Fourier spectrum-based approach to represent and construct decision trees [17, 25], the Collective hierarchical clustering [13] are examples of additional distributed data mining algorithms that can be used with minor modifications for privacy-preserving mining from distributed data. In the recent past, several distributed techniques to mine multi-party data have been reported. A privacy preserving technique to construct decision trees [27] proposed elsewhere [19], multi-party secured computation framework [4], association rule mining from homogeneous [15] and heterogeneous [36] distributed data sets are some examples. There also exists a collection of useful privacy-sensitive data mining primitives such as secure sum computation [30], secure scalar product computation [36].

There is also a somewhat different approach and the algorithms belonging to this group work by first perturbing the data using randomized techniques. The perturbed data is then used to extract the patterns and models. The randomized value distortion technique for learning decision trees [2] and association rule learning [7] are examples of this approach. Additional work on randomized masking of data can be found elsewhere [35].

This paper explores the second approach [2] that works by adding random noise to the data set in order to hide the individual data values of different attributes. It points out that in many cases the noise can be separated from the perturbed data by studying the spectral properties of the data and as a result its privacy can be seriously compromised. Agrawal and Aggarwal [1] consider the approach in [2] and provide an expectation-maximization (EM) algorithm for reconstructing the distribution of the original data from perturbed observations. They also provide information theoretic measures (mutual information) to quantify the amount of privacy provided by a randomization approach. Agrawal and Aggarwal [1] remark that the method suggested in [2] does not take into account the distribution of the original data (which could be used to guess the data value to a higher level of accuracy). However, [1] provides no explicit procedure to reconstruct the original data values. Evfimievski et al. [6, 5] and Rizvi [29] have also considered the approach in [2] in the context of association rule mining and suggest techniques for limiting privacy breaches. Our primary contribution is to provide an explicit filtering procedure, based on random matrix theory, that can be used to estimate the original data values. Before presenting the technique to do that, let us review the randomized value distortion [2] technique in details.

III. Motivation

As noted in the previous section, a growing body of privacy preserving data mining techniques are adopting randomization as a primary tool to "hide" information. While randomization is an important tool, it must be used very carefully in a privacy-preserving application.

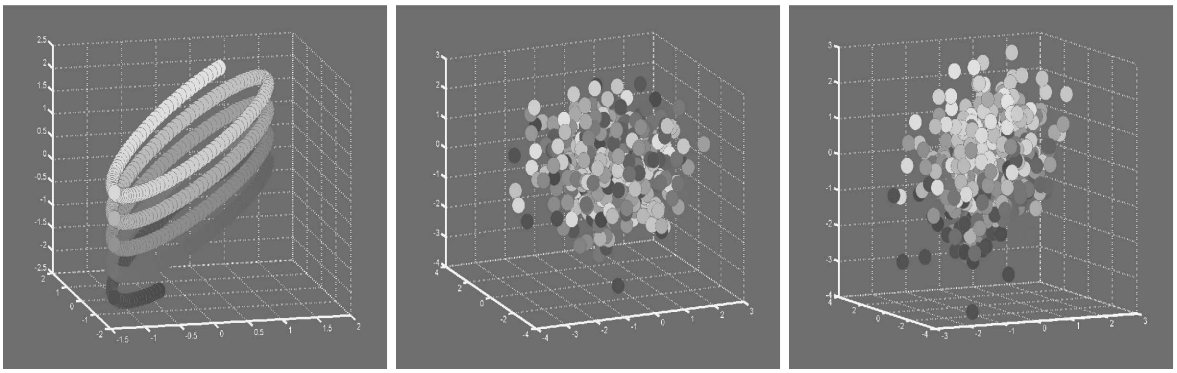

Figure 1: (Left) Non-random data generated in a 3-D space. (Middle) Random noise (uniformly distributed). (Right) Non-random Data with added random noise.

Randomness may not necessarily imply uncertainty. Random events can often be analyzed and their properties can be explained using probabilistic frameworks. Statistics, randomized computation, and many other related fields are full of theorems, laws, and algorithms that rely on probabilistic characterization of random processes that often work quite accurately. The signal processing literature [21] offers many filters to remove white noise from data and they often work reasonably well. Randomly generated structures like graphs demonstrate interesting properties [12]. In short, randomness does seem to have "structure" and this structure may be used to compromise privacy issues unless we pay careful attention.

For example, consider the three dimensional data set shown in Figure 1 (Left). The data is non-random. Figure 1 (Middle) shows a randomly generated data set in the same space with uniform distribution. Figure 1 (Right) shows the perturbed version of the data shown in Figure 1 (Left) after we add the random noise to it. Figures 1 (Left) and (Right) look apparently very different and it may create the perception of privacy protection. However, this may be illusive.

Consider Figure 2 (Left) which shows the spectral properties of the non-random data. The larger sphere is a unit sphere. Smaller spheres represent the different eigenstates. The direction of the position vector of a smaller sphere is given by the associated eigenvector. The magnitude of the vector is the corresponding eigenvalue. Figure 2 (Right) shows the spectral properties of the pure random noise. Note that the eigenvalues of the random noise are all close to 1. This did not happen by chance. In fact, eigenvalues of random noise have some properties that is responsible for this behavior. One can prove that all eigenvalues of a noise matrix with unit variance converges to 1 asymptotically. As we see, the eigenvalues of the randomly generated data are very close to the surface of the unit sphere whereas the non-random data has eigenvalues that are away from the surface. Although, this distinction is not universal, this example illustrates the kind of structure that a linear transformation may expose. We will revisit this property later in this paper and show that under some conditions we may be able to separate the eigenstates of the original sensitive data and the noise using a spectral analysis of the perturbed "privacy-protected" data. This in turn allows us to reconstruct the original sensitive data. Properties like this offer challenges to the designers of privacy preserving data mining algorithms that use randomization as a tool to hide the sensitive data.

The rest of this paper illustrates this challenge in the context of a well-known privacy preserving technique that works using random additive noise. The following section first explains this random value perturbation approach.

IV. Random Value Perturbation Technique: A Brief Review

For the sake of completeness, we now briefly review the random data perturbation method suggested in [2] for hiding the data (i.e. guaranteeing protection against the reconstruction of the data) while still being able to estimate the underlying distribution. We also discuss the procedure for reconstructing the original density function, as suggested in [2].

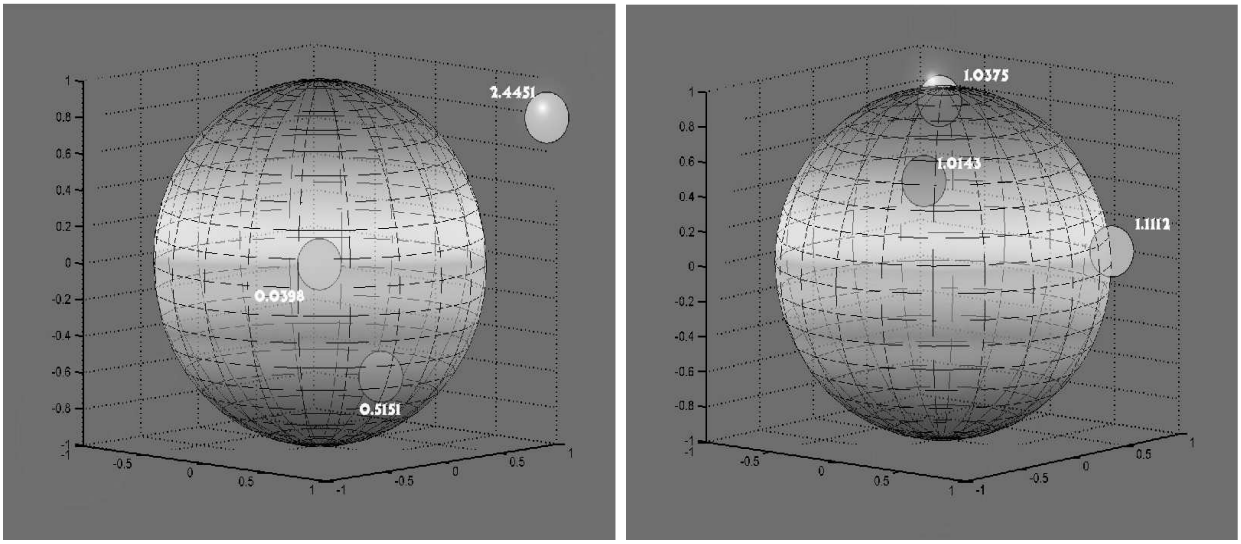

Figure 2: (Left) Eigenstates of the data. The larger sphere is a unit sphere. Smaller spheres represent the different eigenstates. The direction of the position vector of a smaller sphere is given by the associated eigenvector. The magnitude of the vector is the corresponding eigenvalue. (Right) Eigenstates of the random noise.

A. Perturbing the Data

The random value perturbation method attempts to preserve privacy of the data by modifying values of the sensitive attributes using a randomized process [2]. The authors explore two possible approaches — Value-Class Membership and Value Distortion — and emphasize the Value Distortion approach. In this approach, the owner of a dataset returns a value $u_i + v$, where u_i is the original data, and v is a random value drawn from a certain distribution. Most commonly used distributions are the uniform distribution over an interval $[-\alpha, \alpha]$ and Gaussian distribution with mean $\mu = 0$ and standard deviation σ. The n original data values $u_1, u_2, \ldots, u_n$ are viewed as realizations of n independent and identically distributed (i.i.d.) random variables U_i, $i = 1, 2, \ldots, n$, each with the same distribution as that of a random variable U. In order to perturb the data, n independent samples $v_1, v_2, \ldots, v_n$, are drawn from a distribution V. The owner of the data provides the perturbed values $u_1 + v_1, u_2 + v_2, \ldots, u_n + v_n$ and the cumulative distribution function $F_V(r)$ of V. The reconstruction problem is to estimate the distribution $F_U(x)$ of the original data, from the perturbed data.

B. Estimation of Distribution Function from the Perturbed Dataset

The authors [2] suggest the following method to estimate the distribution $F_U(u)$ of U, given n independent samples $w_i = u_i + v_i$, $i = 1, 2, \ldots, n$ and $F_V(v)$. Using Bayes' rule, the posterior distribution function $F'_U(u)$ of U, given that $U + V = w$, can be written as

$$F'_U(u) = \frac{\int_{-\infty}^{u} f_V(w - z) f_U(z) dz}{\int_{-\infty}^{\infty} f_V(w - z) f_U(z) dz},$$

which upon differentiation with respect to u yields the density function

$$f'_U(u) = \frac{f_V(w - u) f_U(u)}{\int_{-\infty}^{\infty} f_V(w - z) f_U(z) dz},$$

where $f_U(.)$, $f_V(.)$ denote the probability density function of U and V respectively. If we have n independent samples $u_i + v_i = w_i$, $i = 1, 2, \ldots, n$, the corresponding posterior distribution can be obtained by averaging:

$$f'_U(u) = \frac{1}{n} \sum_{i=1}^{n} \frac{f_V(w_i - u) f_U(u)}{\int_{-\infty}^{\infty} f_V(w_i - z) f_U(z) dz}. \quad (1)$$

For sufficiently large number of samples n, we expect the above density function to be close to the real density function $f_U(u)$. In practice, since the true density $f_U(u)$ is unknown, we need to modify the right-hand side of equation (1). The authors suggest an iterative procedure where at each step $j = 1, 2, \ldots$, the posterior density $f_U^{j-1}(u)$ estimated at step $j-1$ is used in the right-hand side of equation (1). The uniform density is used to initialize the iterations. The iterations are carried out until the difference between successive estimates becomes small. In order to speed up computations, the authors also discuss approximations to the above procedure using partitioning of the domain of data values.

V. Randomness and Patterns

The random perturbation technique "apparently" distorts the sensitive attribute values and still allows estimation of the underlying distribution information. However, does this apparent distortion fundamentally prohibit us from extracting the hidden information? In this section we explore this question. We use a particular filtering technique that is designed based on properties of random matrices. This section presents a discussion on the properties of random matrices and presents some results that will be used later in this paper.

Random matrices [23] also exhibit many interesting properties that are often exploited in high energy physics [23], signal processing [32], and even data mining [18]. The random noise added to the data can be viewed as a random matrix and therefore its properties can be understood by studying the properties of random matrices. In this paper we shall develop a spectral filter designed based on random matrix theory for extracting the hidden data from the data perturbed by random noise. Our filtering approach is based on the observation that the distribution of eigenvalues of random matrices [23] exhibit some well known characteristics. The rest of this section discusses some of the important spectral properties of random matrices.

A random matrix is a matrix whose elements are random variables with given probability laws. The theory of random matrices deals with the statistical properties of the eigenvalues of such matrices. Eigenvalues of random matrices offer many interesting properties. For example, Wigner's semi-circle law [38], which says if V is an $n \times n$ matrix and has i.i.d. entries with zero mean and unit variance, the distribution of eigenvalues of $\frac{V+V'}{2\sqrt{2n}}$ has a probability density function given by

$$f(x) = \begin{cases} \frac{1}{\pi}(2n - x^2)^{1/2}, & |x| < \sqrt{2n} \\ 0, & \text{otherwise.} \end{cases}$$

In this paper, we are mainly concerned about distribution of eigenvalues of the sample covariance matrix obtained from a random matrix. Let V be a random $m \times n$ matrix whose entries are V_{ij}, $i = 1, \ldots, m$, $j = 1, \ldots, n$, are i.i.d. random variables with zero mean and variance σ^2. The covariance matrix of X is given by $Y = \frac{1}{m}V'V$. Clearly, Y is an $n \times n$ matrix. Let $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ be the eigenvalues of Y. Let

$$F_n(x) = \frac{1}{n}\sum_{i=1}^{n} U(x - \lambda_i),$$

be the empirical cumulative distribution function (c.d.f.) of the eigenvalues λ_i, $(1 \leq i \leq n)$, where

$$U(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}$$

is the unit step function. In order to consider the asymptotic properties of the c.d.f. $F_n(x)$, we will consider the dimensions $m = m(N)$ and $n = n(N)$ of matrix X to be functions of a variable N. We will consider asymptotics such that in the limit as $N \to \infty$, we have $m(N) \to \infty$, $n(N) \to \infty$, and $\frac{m(N)}{n(N)} \to Q$, where $Q \geq 1$. Under these assumptions, it can be shown that [14] the empirical c.d.f. $F_n(x)$ converges in probability to a continuous distribution function $F_Q(x)$ for every x, whose probability density function (p.d.f.) is given by

$$f_Q(x) = \begin{cases} \frac{Q\sqrt{(x-\lambda_{\min})(\lambda_{\max}-x)}}{2\pi\sigma^2 x} & \lambda_{\min} < x < \lambda_{\max} \\ 0 & \text{otherwise,} \end{cases} \tag{2}$$

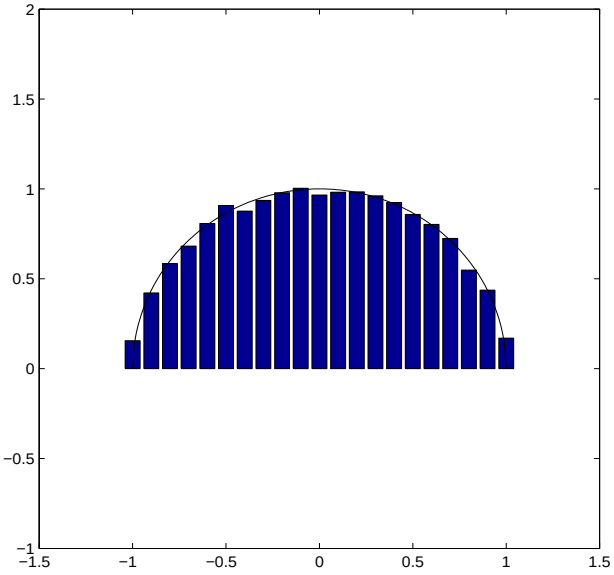

Figure 3: Wigner's semi-circle law: Distribution of the eigenvalues of $\frac{V+V'}{2\sqrt{2n}}$ where V is a random matrix. It takes the shape of a semi-circle.

where $\lambda_{\min}$ and $\lambda_{\max}$ are as follows:

$$\begin{aligned} \lambda_{\min} &= \sigma^2(1 - 1/\sqrt{Q})^2. \\ \lambda_{\max} &= \sigma^2(1 + 1/\sqrt{Q})^2. \end{aligned} \tag{3}$$

Further refinements of this result and other discussions can be found in [32, 9, 22, 3, 8, 39, 31].

VI. Separating the Data from the Noise

Consider an $m \times n$ data matrix U and a noise matrix V with same dimensions. The random value perturbation technique generates a modified data matrix $U_p = U + V$. Our objective is to extract U from U_p. Although the noise matrix V may introduce seemingly significant difference between U and U_p, it may not be successful in hiding the data.

Random noise has well defined probabilistic properties that may be used to identify the noise component of the perturbed data matrix U_p in an appropriate representation. The rest of this paper argues that the spectral representation of the data allows us to do exactly that.

Consider the covariance matrix of U_p:

$$\begin{aligned} U_p^T U_p &= (U + V)^T (U + V) \\ &= U^T U + V^T U + U^T V + V^T V. \end{aligned} \tag{4}$$

Now note that when the signal random vector (rows of U) and noise random vector (rows of V) are uncorrelated, we have $E[U^T V] = E[V^T U] = 0$. The uncorrelated assumption is valid in practice since the noise V that is added to the data U is generated by a statistically independent process. Recall that the random value perturbation technique discussed in the previous section introduces uncorrelated noise to hide the signal or the data. If the number of observations is sufficiently large, we have that $U^T V \sim 0$ and $V^T U \sim 0$. Equation 4 can now be simplified as follows:

$$U_p^T U_p = U^T U + V^T V \tag{5}$$

Since the correlation matrices U^TU, $U_p^TU_p$, and V^TV are symmetric and positive semi-definite, let

$$\begin{aligned} U^TU &= Q_u\Lambda_uQ_u^T, \\ U_p^TU_p &= Q_p\Lambda_pQ_p^T, \text{ and} \\ V^TV &= Q_v\Lambda_vQ_v^T, \end{aligned} \tag{6}$$

where Q_u, Q_p, Q_v are orthogonal matrices whose column vectors are eigenvectors of U^TU, $U_p^TU_p$, V^TV, respectively, and Λ_u, Λ_p, Λ_v are diagonal matrices with the corresponding eigenvalues on their diagonals.

The following result from matrix perturbation theory [37] gives a relationship between Λ_u, Λ_v, and Λ_p.

Theorem 1: [37] Suppose $\lambda_{1,(a)} \geq \lambda_{2,(a)} \geq \ldots \lambda_{n,(a)} \geq 0$, $a \in \{u, p, v\}$ are the eigenvalues of U^TU, $U_p^TU_p$, and V^TV, respectively. Then, for $i = 1, \ldots, n$,

$$\lambda_{i,(p)} \in [\lambda_{i,(u)} + \lambda_{n,(v)}, \lambda_{i,(u)} + \lambda_{1,(v)}].$$

This theorem provides us a bound on the change in the eigenvalues of the data correlation matrix U^TU in terms of the minimum and maximum eigenvalues of the noise correlation matrix V^TV. Now let us take a step further and explore the properties of the eigenvalues of the perturbed data matrix U_p for large values of m.

Lemma 1: Let data matrix U and noise matrix V be of size $m \times n$ and $U_p = U + V$. Let Q_u, Q_p, Q_v be orthogonal matrices and Λ_u, Λ_p, Λ_v be diagonal matrices as defined in equation (6). If $m/n \to \infty$ then $\Lambda_p = \Delta\Lambda_u\Delta^T + \Lambda_v$ where $\Delta = Q_p^TQ_u$.

Proof:
Using Equations 5 and 6 we can write,

$$\begin{aligned} Q_p\Lambda_pQ_p^T &= Q_u\Lambda_uQ_u^T + Q_v\Lambda_vQ_v^T \\ \Rightarrow \Lambda_p &= Q_p^TQ_u\Lambda_uQ_u^TQ_p + Q_p^TQ_v\Lambda_vQ_v^TQ_p \\ &= \Delta\Lambda_u\Delta^T + Q_p^TQ_v\Lambda_vQ_v^TQ_p \end{aligned} \tag{7}$$

Let the minimum and maximum eigenvalues of V be $\lambda_{\min,(v)}$ and $\lambda_{\max,(v)}$ respectively. It follows from equations (2) and (3) that $m/n \to \infty$ all the eigenvalues in Λ_v become identical since $\lim_{m/n=Q\to\infty} \lambda_{\max,(v)} = \lim_{m/n=Q\to\infty} \lambda_{\min,(v)} = \sigma^2$ (say). This implies that, as $m/n \to \infty$, $\Lambda_v \to \sigma^2 I$, where I is the $n \times n$ identity matrix. Therefore, if the number of observations m is large enough (note that, in practice, number of features n is fixed), $V^TV = Q_v\Lambda_vQ_v^T = \sigma^2Q_vQ_V^T = \sigma^2 I$. Therefore equation (7) becomes

$$\begin{aligned} \Lambda_p &= \Delta\Lambda_u\Delta^T + Q_p^TQ_p\Lambda_vQ_p^TQ_p \\ \Lambda_p &= \Delta\Lambda_u\Delta^T + \Lambda_v. \end{aligned} \tag{8}$$

■

If the norm of the perturbation matrix V is small, the eigenvectors Q_p of $U_p^TU_p$ would be close to the eigenvectors $Q_u^TQ_u$ of U^TU. Indeed, matrix perturbation theory provides precise bounds on the angle between eigenvectors (and invariant subspaces) of a matrix U and that of its perturbation $U_p = U + V$, in terms of the norms of the perturbation matrix V. For example, let (x_u, λ_u) be an eigenvector-eigenvalue pair for matrix U^TU and $\epsilon = \|V^TV\|_2 = \sigma_{\max}(V^TV)$ be the two-norm of the perturbation, where $\sigma_{\max}(V^TV)$ is the largest singular value of V^TV. Then there exists an eigenvalue-eigenvector pair (x_p, λ_p) of $U_p^TU_p$ satisfying [37, 33]

$$\tan(\angle(x_u, x_p)) < 2\frac{\epsilon}{\delta - \epsilon},$$

where δ is the distance between λ_u and the closest eigenvalue of U^TU, provided $\epsilon < \delta$. This shows that the eigenvalues of U^TU and $U_P^TU_P$ are in general close, for small perturbations. Moreover,

$$|\lambda_u - x^*U_px_u| < 2\frac{\epsilon^2}{\delta - \epsilon},$$

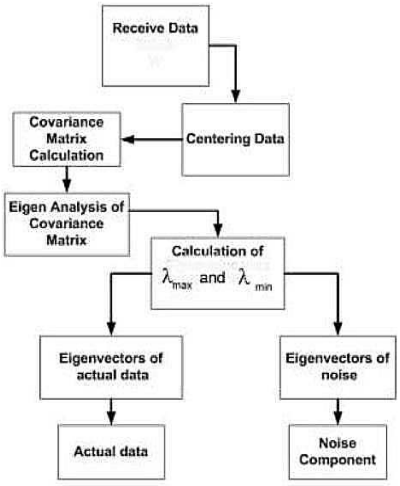

Figure 4: Flowchart of the spectral filtering technique.

where x^* is the conjugate-transpose of x. Consequently, the product $\Delta = Q_p^T Q_u$, which is the matrix of inner products between the eigenvectors of $U^T U$ and $U_p^T U_p$ would be close to an identity matrix; i.e., $\Delta = Q_p^T Q_u \approx I$. Thus equation 8 becomes

$$\Lambda_p \approx \Lambda_u + \Lambda_v. \tag{9}$$

Suppose the signal covariance matrix has only a few dominant eigenvalues, say $\lambda_{1,(u)} \geq \cdots \geq \lambda_{k,(u)}$, with $\lambda_{i,(u)} \leq \epsilon$ for some small value ϵ and $i = k+1, \ldots, n$. This condition is true for many real-world signals. Suppose $\lambda_{k,(u)} > \lambda_{1,(v)}$, the largest eigenvalue of the noise covariance matrix. It is then clear that we can separate the signal and noise eigenvalues Λ_u, Λ_v from the eigenvalues Λ_p of the observed data by a simple thresholding at $\lambda_{1,(v)}$.

Note that equation (9) is only an approximation. However, in practice, one can design a filter based on this approximation to filter out the perturbation from the data. Experimental results presented in the following sections indicate that this provides a good recovery of the data in many cases.

VII. Random Matrix-Based Data Filtering

This section describes the proposed filter for extracting the original data from the noisy perturbed data. Suppose actual data U is perturbed by a randomly generated noise matrix V in order to produce $U_p = U + V$. Let $u_{p,i} = \mathbf{u}_i + \mathbf{v}_i$, $i = 1, 2, \ldots, m$, be m (perturbed) data points, each being a vector of n features.

The proposed filtering technique first calculates the covariance matrix of the perturbed data U_p. Using the distribution of eigenvalues of the covariance matrix, and the theory of random matrices, the covariance matrix of U_p is decomposed into the noise and the signal part. The eigenstates corresponding to signal part are then used to reconstruct the actual data. This can be done by simply projecting the data along the signal eigenvectors and then mapping it back to the original space.

In the following section, we discuss the proposed filtering procedure. We first explore the case where the distribution $F_V(v)$ of the random noise V (including the variance) is known, as required by the random value perturbation scheme [2]. Next we discuss how the noise variance can be estimated from the eigenvalue distribution of the perturbed data. The reader should note that the random value perturbation scheme provides information about the noise distribution. So estimation of the noise variance is *not necessary*. We explored that case in order to develop a broader understanding about the performance of the proposed filtering technique.

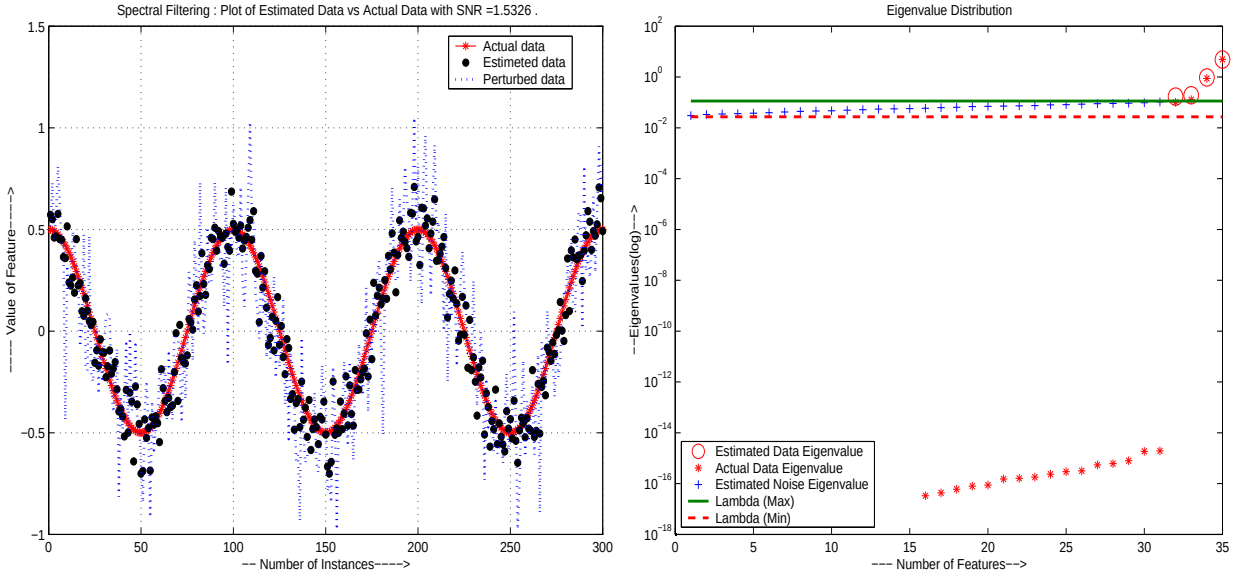

Figure 5: (Left) Estimation of original sinusoidal data with known random noise variance. (Right) Distribution of eigenvalues of actual data, and estimated eigenvalues of random noise and actual data.

A. Known Noise Variance

When the noise distribution $F_V(v)$ of V is completely known (as required by the random value perturbation technique [2]), the noise variance σ^2 is first calculated from the given distribution. Equation 2 is then used to calculate λ_{max} and λ_{min} which provide the theoretical bounds of the eigenvalues corresponding to noise matrix V. From the perturbed data, we compute the eigenvalues of its covariance matrix Y, say $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$. Then we identify the noisy eigenstates $\lambda_i \leq \lambda_{i+1} \leq \cdots \leq \lambda_j$ such that $\lambda_i \geq \lambda_{min}$ and $\lambda_j \leq \lambda_{max}$. The remaining eigenstates are the eigenstates corresponding to actual data. Let, Λ_v = diag $(\lambda_i, \lambda_{i+1}, \ldots, \lambda_j)$ be the diagonal matrix with all noise-related eigenvalues, and A_v be the matrix whose columns are the corresponding eigenvectors. Similarly, let Λ_u be the eigenvalue matrix for the actual data part and A_u be the corresponding eigenvector matrix which is an $n \times k$ matrix ($k \leq n$). Based on these matrices, we decompose the covariance matrix Y into two parts, Y_s and Y_r with $Y = Y_s + Y_r$, where $Y_r = A_v \Lambda_r A_v^T$, is the covariance matrix corresponding to random noise part, and $Y_s = A_u \Lambda_u A_u^T$, is the covariance matrix corresponding to actual data part. An estimate $\hat{U}$ of the actual data U is obtained by projecting the data U_p on to the subspace spanned by the columns of A_u. In other words, $\hat{U} = U_p A_u A_u^T$.

B. Unknown Noise Variance

Using random matrix theory as described in section V, the spectral filtering technique can even be extended to estimate the actual data when the noise variance σ^2 is not known. For this, the noise variance is estimated first using the perturbed data and that estimated noise variance is then used to filter the perturbed data. In order to estimate the noise variance σ^2 we first compute the eigenvalues of the covariance matrix Y of the perturbed data W. A histogram of the eigenvalue distribution is plotted and compared to that of the theoretical noise eigenvalue density function $f_Q(x)$ given in equation (2). Note that the density function $f_Q(x)$ depends on the variance σ^2. Typically, the theoretical density function $f_Q(x)$ is a good fit to the left portion of the histogram of the computed eigenvalues, corresponding to small eigenvalues. The larger eigenvalues that do not fit this theoretical density function correspond to the actual information part of the perturbed data. An iterative procedure is employed to obtain the value of σ that results in the best fit of $f_Q(x)$ to the observed histogram.

VIII. Experimental Results

In this section, we present results of our experiments with the proposed spectral filtering technique. We have compared the performance of our filtering technique with two other common filters used in the literature.

This section also includes discussion on the effect of noise variance on the performance of the spectral filtering method.

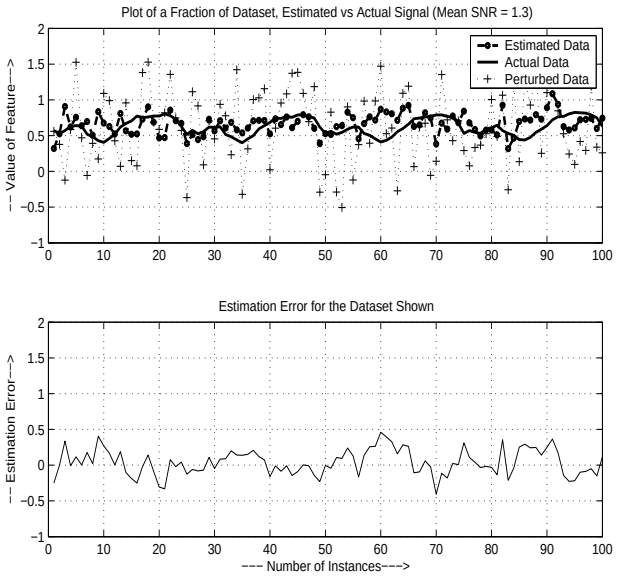

Figure 6: Spectral filtering used to estimate real world audio data. Waveform of a audio signal is estimated from its perturbed version with high accuracy.

A. Estimation with Known Perturbing Distribution

We tested our privacy breaching technique using several datasets of different sizes. We considered both artificially generated and real data sets. Towards that end, we generated a dataset with 35 features and 300 instances. Each feature has a specific trend like sinusoidal, square, and triangular shape. The actual dataset is perturbed by adding Gaussian noise (with zero mean and known variance), and our proposed technique is applied to recover the actual data from the perturbed data. Figure 5 (Left) shows the result of our spectral filtering for one such feature where the actual data has a sinusoidal trend. The filtering technique appears to provide an close estimate of the individual values of the actual data. Figure 5 (Right) shows the distribution of eigenvalues of the actual and perturbed data. It also identifies the estimated noise eigenvalues and the theoretical bounds $\lambda_{\max}$ and $\lambda_{\min}$. As we see, the filtering method accurately distinguishes between noisy eigenvalues and eigenvalues corresponding to actual data. Note that the estimated eigenvalues of actual data is very close to eigenvalues of actual data and almost overlap with them above $\lambda_{\max}$. The eigenvalues of actual data below $\lambda_{\min}$ are practically negligible. Thus, the estimated eigenvalues of the actual data capture most of the information and discard the additive noise.

The random matrix-based filtering technique can also be extended to datasets with a single feature, i.e when the dataset is a single column vector. The data vector is perturbed with a noise vector with the same dimension. The perturbed data vector is then split into a fixed number of vectors with equal length and all of these vectors are appended to form a matrix. The spectral filtering technique is then applied to this matrix to estimate the original data. After the data matrix is estimated, its columns are concatenated to form a single vector.

We used a real world single feature data set to verify the performance of the spectral filtering. The dataset used is the scaled amplitude of the waveform of an audio tune recorded using a fixed sampling frequency. The tune recorded is fairly noise free with $10,000$ sample points. We perturbed this data with additive Gaussian noise.

We define the term *Signal-to-Noise Ratio* (SNR) to quantify the relative amount of noise added to actual data to perturb it:

$$\text{SNR} = \frac{\text{Variance of Actual Data}}{\text{Noise Variance}}. \tag{10}$$

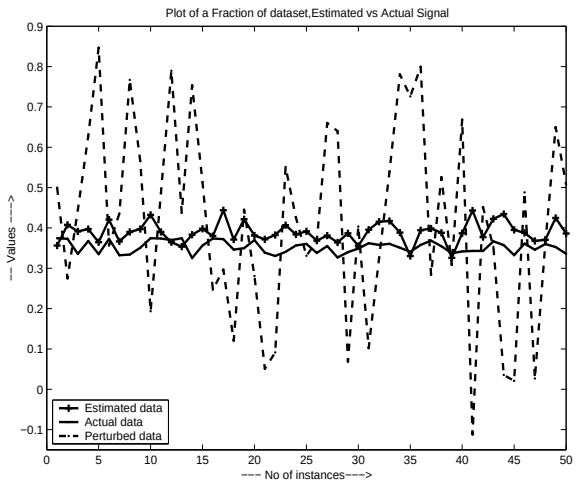

Figure 7: Plot of the individual values of a fraction of the dataset with 'Triangular' distribution. Spectral filtering gives close estimation of individual values.

In this experiment, the noise variance was chosen to yield a signal-to-noise ratio of 1.3. We split this vector of perturbed data into 40 columns, each containing 250 points, and applied the spectral filtering technique to recover the actual data. The result is shown in Figure 6. For the sake of clarity, only a fraction of dataset is shown, and estimation error is plotted for that fraction. As shown in Figure 6, the perturbed data is very different from the actual data, whereas the estimated data is a close approximation of the actual data. The estimation performance is similar to that for a multi-featured data (see Figure 5 (Left)).

B. Comparison With Results in [2]

The proposed spectral filtering technique can estimate values of individual data-points from the perturbed dataset. This point-wise estimation can then be used to reconstruct the distribution of actual data as well. The methods suggested by [2, 1] can only reconstruct the distribution of the original data from the data perturbed by random value distortion; but it does not consider estimation of the individual values of the data-points. The spectral filtering technique, on the other hand, is explicitly designed to reconstruct the individual data-points and hence, also the distribution of the actual dataset.

We tried to replicate the experiment reported in [2] using our method to recover the triangular distribution. We used a vector data of $10,000$ values from a triangular distribution as shown in Figure 2 in [2]. The individual values of actual data are between 0 and 1. The dataset was generated by dividing the interval $[0, 1]$ into a number of sub-intervals and generating a proportionate number of samples, uniformly distributed in each sub-interval. The sub-intervals were sorted, although the samples inside each sun-interval were not sorted. This introduces correlations between the columns when the vector data was split into 50 columns (see description below). We added Gaussian noise with mean 0 and standard deviation $\sigma = 0.25$ to this data and split the data vector into 50 columns, each having 200 values. We then applied our spectral filter to recover the actual data from the perturbed data. Figure 7 shows a portion of the actual data, their values after distortion, and their estimated values. Note that the estimated values are very close to the actual values, compared to the perturbed values. Using the estimate of individual data-points, we reconstruct the distribution of the actual data. Figure 8 (Left) shows estimation of the distribution from the estimated value of individual data-points. The distribution of the perturbed data is very different than the actual triangular distribution, but the estimated distribution looks very similar to the original distribution. Figure 8 (Right) shows the error in estimation of the actual data for the whole dataset ($10,000$ points). The estimation error remains within ± 0.25 in this experiment. These results show that our method recovers the original distribution quite accurately along with individual data-points, similar to the result reported in [2]. [2] claims that privacy is preserved by random perturbation because their method can reconstruct only the distribution of actual data from the perturbed version, while spectral filtering can filter out the individual values of actual data closely from perturbed data, thus questioning reliability of the randomized data perturbation technique

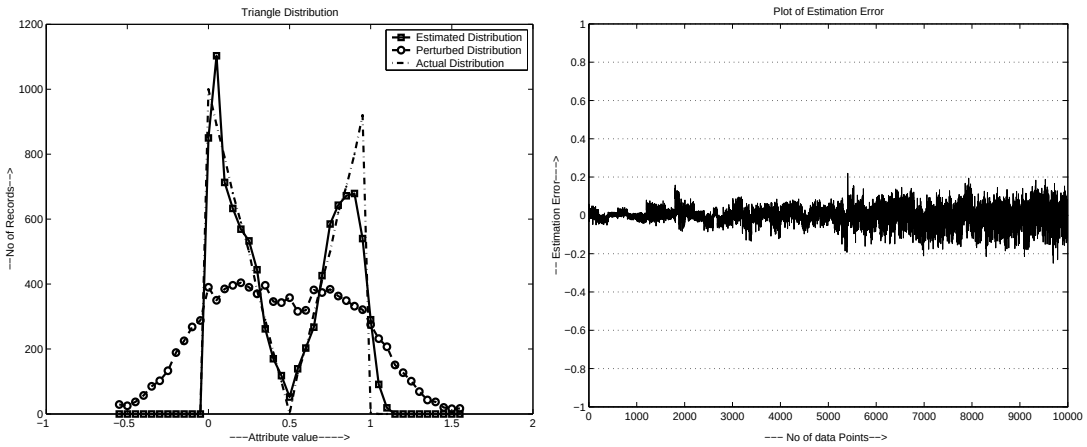

Figure 8: (Left) Reconstruction of the 'Triangular' distribution. Perturbed data distribution does not look like a triangular distribution, but reconstructed distribution using spectral filtering resembles the original distribution closely. (Right) Estimation error for the experiment reported in the left figure. The error is within ± 0.25.

as a privacy preserving tool.

C. Comparison with Moving Average and Weiner Filtering Techniques

The spectral filtering technique considered in this paper is certainly not the only filter that can be used to removing noise. In this section we report the relative performance of the spectral technique with respect to few other existing filters.

We compared the performance of the proposed spectral filtering technique with two other well known noise filtering techniques, viz. moving average filtering and Wiener filtering [24]. These two techniques are well-known and widely used in signal processing to filter out noise. Moving average (MA) filter simply computes the average of the perturbed data over a window of fixed size, centered around each data point, whereas the Wiener filter uses information about the power spectral density (equivalently, autocorrelation) of the data and noise. We applied MA, Wiener, and Spectral filter on the same dataset perturbed with random noise and in most cases we observed that spectral filtering performs better than the MA and Wiener filters. Figure 9 compares the performance of spectral filtering with that of the MA filter. The estimation accuracy of the spectral filter is clearly higher than that of the MA filter for the given dataset which has a square trend in its values. The window size used in this experiment is 10. Figure 10 shows the comparison with respect to the Wiener filter. Clearly, for the given dataset, spectral filter does a better job compared to Wiener filter.

D. Comparison with Principal Component Analysis (PCA) Based Filtering Technique

In this section we compare the performance of our spectral filtering technique with PCA based filtering. We also analyze the effect of batch number (Q in equation (2)) on the estimation error and report experiments run in batch mode.

PCA is a traditional statistical technique based on eigenanalysis of data. In this method the sample covariance matrix of the data is calculated and its eigenstates are evaluated. Then principal eigenvalues are separated out that capture 90% and 75% of overall eigenvalues. The corresponding principal eigenvectors are used to obtain an estimation of the observed data as we do in our spectral filtering.

In our experiment, we used an artificial dataset which has $n = 100$ features, and 20,000 observations. We started with first 100 observations in a batch, and kept on adding 100 more observations in each run of the experiment so that the ratio $Q = m/n$, with m being the number of observations obtained upto the current batch, increases from 1. Experimental results for 20 batches at SNR=1.03 are shown in Figure 11. The top figure shows the error in the estimated covariance matrix compared with true data covariance matrix, as a function of batch number using spectral filtering technique, traditional PCA based method with eigenvectors capturing 90% and 75% of the total variance and a simple moving average filter with the window size 100.

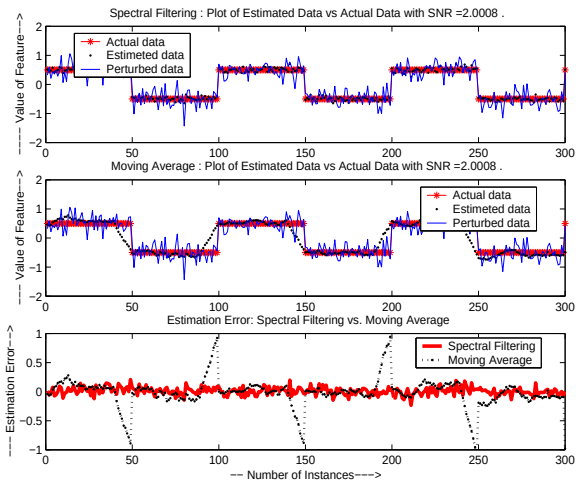

Figure 9: Relative performance of the spectral filtering technique with respect to a moving average filter. Spectral filtering offers better filtering performance compared to the Moving average filter in this case.

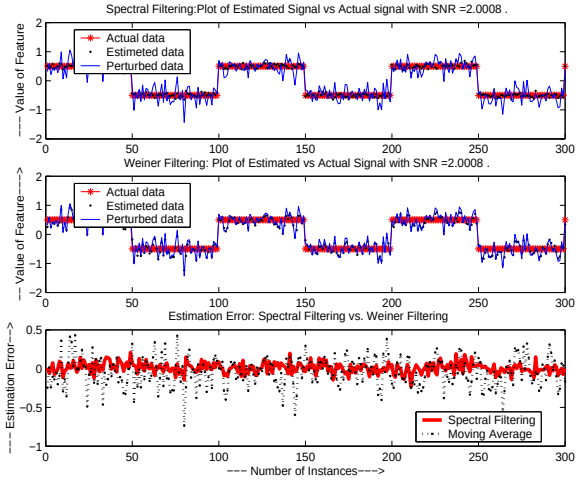

Figure 10: Relative performance of Spectral filtering technique with respect to a Weiner filter. Spectral filtering offers better filtering performance compared to the Weiner filter in this case.

The bottom figure depicts the mean square error in estimated data for all of these filtering methods as a function of batch number. Note that after 6 or 7 batches, filtering errors reduce to relatively stable values. From the figure, it is clear that our spectral filter provides the best noise filtering in terms of mean-squared error.

E. Effect of Perturbation Variance and the Inherent Random Component of the Actual Data

Quality of the data recovery depends upon the relative noise content in the perturbed data. We use the SNR (see equation (10)) to quantify the relative amount of noise added to actual data to perturb it. As the noise added to the actual value increases, the SNR decreases. Our experiments show that the proposed filtering method predicts the actual data reasonably well up to a SNR value of 1.0 (i.e. 100% noise). The results shown in Figure 5 (Left) corresponds to an SNR value nearly 2, i.e. noise content is about 50%. Figure 7 shows a data-block where the SNR is 1.9. As the SNR goes below 1, the estimation becomes too erroneous. Figure 12 (Left) shows the difference in estimation accuracy as the SNR increases from 1. The dataset used here has a sinusoidal trend in its values. The top graph corresponds to 23% noise (SNR = 4.3), whereas the bottom graph corresponds to 100% noise (SNR = 1.0). Figure 12 (Right) shows the variation

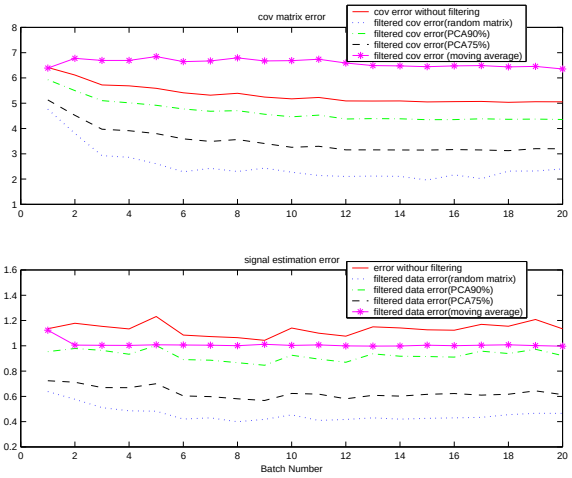

Figure 11: Error in covariance matrix estimation and data filtering for PCA based filtering, spectral filtering and moving average

of estimation error with SNR. As the SNR value decreases, mean error in estimation shows an increasing trend.

Another important factor that affects the quality of recovery of the actual data is the inherent noise in the actual dataset (apart from the perturbation noise added intentionally). If the actual dataset has a random component in it, and random noise is added to perturb it, spectral filtering method does not filter the actual data accurately. Our experiments with some inherently noisy real life dataset show that the eigenvalues of signal and noise no longer remains clearly separable since the their eigenvalues may not be distributed over two non-overlapping regimes any longer.

For dataset with specific trend like the one shown in Figure 5 (Left), due to absence of any random component in actual data, equation (9) holds closely, giving an accurate estimate of the actual data. However, the accuracy deteriorates when the original data set contains random noise in itself.

We performed experiments with artificial dataset with specific trend in its value as well as real world dataset containing a random component. Figure 13 (Left) shows that our method gives a close estimation of actual data when the dataset has specific trend (sinusoid), and SNR of the perturbed data is 1.1. We also applied our method to "Ionosphere data" available from [28], which is inherently noisy. We perturbed the original data with random noise such that mean SNR is same as the artificial dataset, i.e. 1.1. Figure 13 (Right) shows that recovery quality is poor compared to datasets having definite trend.

However, this opens up a different question: Is the random component of the original data set really important as far as data mining is concerned? One may argue that most data mining techniques exploit only the non-random structured patterns of the data. Therefore, losing the inherent random component of the original data may not be important in a privacy preserving data mining application.

F. Estimation with Unknown Perturbing Noise Distribution

Spectral filtering technique can be extended to estimate actual dataset even when distribution of noise added to perturb the actual dataset is not known. In such case , we can use the random matrix theory described in section V to estimate the noise variance first. From the eigenvalues of covariance matrix of actual data, a histogram of the eigenvalue distribution is obtained, and this is compared with best possible theoretical density function given by equation (2). The variance corresponding to the best fit gives the estimation of the noise variance.

To get the best estimation of variance, the algorithm estimates noise variance from the best fit curve several times. In each trial , the variance estimation algorithm starts with a very small variance value near zero, create the theoretically generated distribution and measures the mean square error between it and histogram of eigenvalues of actual data. It then increases variance by a small value, again computes the mean square error and compares it with the previous error to get the minimum error and corresponding

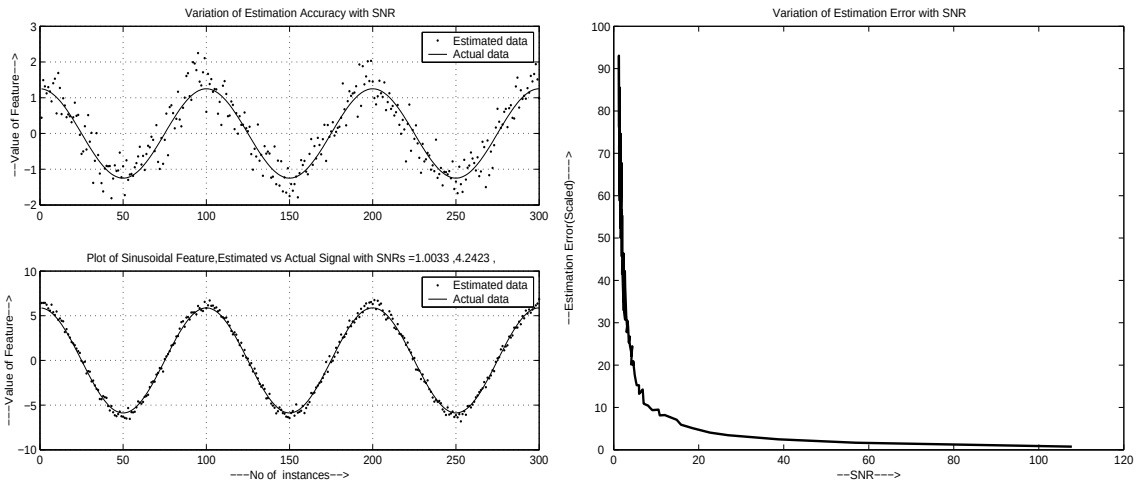

Figure 12: (Left) A higher noise content (low SNR)leads to less accurate estimation. SNR in the left figure is 1, while that for the right figure is 4.3. (Right) Variation of mean estimation error with change in SNR values. As SNR decreases, mean estimation error increases.

variance. The algorithm performs the operation up-to a threshold value of variance, and stores of the variance corresponding to minimum mean square error between theoretically generated density function curve and histogram of eigenvalues of actual data. That value of variance is treated as the estimated value of noise variance for that particular trial. In our experiment, we used 100 such trials for each variance estimation. After the set of estimates are calculated from all trials, the distribution of estimated variances is checked for outliers in them. The mean μ_1 and standard deviation σ_1 of the estimates are calculated, and values lying outside the regime depicted by $\mu_1 \pm 2\sigma_1$ are discarded. During each trial, if the algorithm does not get best fit within a predefined threshold value of variance, it stores that threshold value of variance as the estimation. These values are also treated as outliers at the end and are discarded.

After discarding the outlier estimations, an average of the rest of the estimates are taken to get the actual estimate of noise variance. We have noticed that discarding the outliers and taking average of the remaining number of estimate improves the estimation accuracy to a large extent. Once the noise variance is estimated, the same technique is applied as before to estimate the original data. Figure 14 (Left) shows the estimation of actual data having 300 values and a sawtooth trend with SNR value of 4.25 when distribution of noise is not known. The average over 100 estimates of noise variance after discarding the outliers gave an estimated variance of 0.83452 where the actual noise variance is 0.85. Although not all the estimates are always so close, on an average, the difference between the estimated variance and true variance remains within 10% of the actual variance in all our experiments. Figure 14 (Right) shows the corresponding distribution of eigenvalues of actual data, estimated noise and estimated data.

IX. Conclusion and Future Work

Preserving privacy in data mining activities is a very important issue in many applications. Randomization-based techniques are likely to play an important role in this domain. However, this paper illustrates some of the challenges that these techniques face in preserving the data privacy. It showed that under certain conditions it is relatively easy to breach the privacy protection offered by the random perturbation based techniques. It provided extensive experimental results with different types of data and showed that this is really a concern that we must address. In addition to raising this concern, the paper makes several other contributions that are discussed later in this section.

The proposed technique works based on the spectral properties of the covariance matrix. The covariance matrix is a diagonal matrix when the data columns are completely uncorrelated. This paper presented some experimental results using univariate data ($m \times 1$ dimensional matrix) where the covariance matrix is computed from the multi-dimensional data ($\frac{m}{k} \times k$ dimensional matrix) where a column corresponds to one of the k consecutive sub-sequences with $\frac{m}{k}$ entries. If the observations in this univariate data are mutually

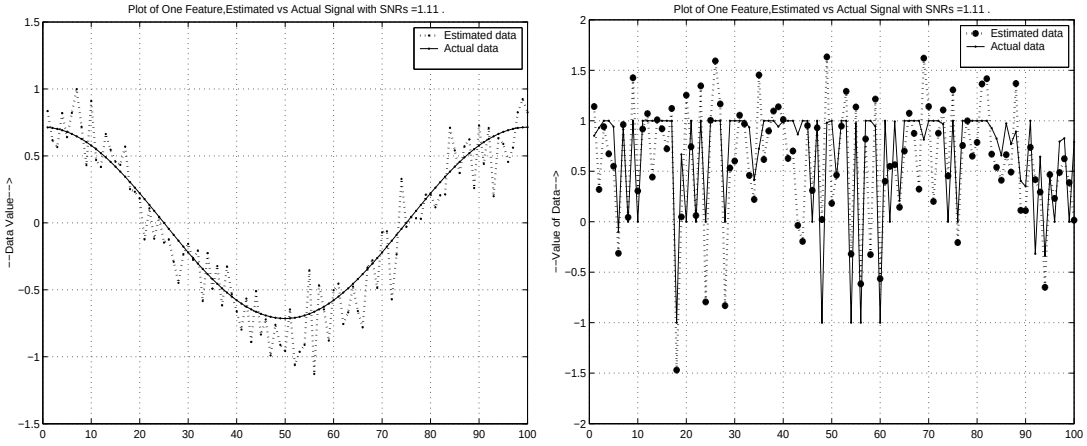

Figure 13: (Left) The spectral filtering technique recovers actual data with specific trend in its value closely. (Right) It performs poorly on a dataset with a random component in its actual value. However, it is not clear if losing the random component of the data is a concern for data mining applications.

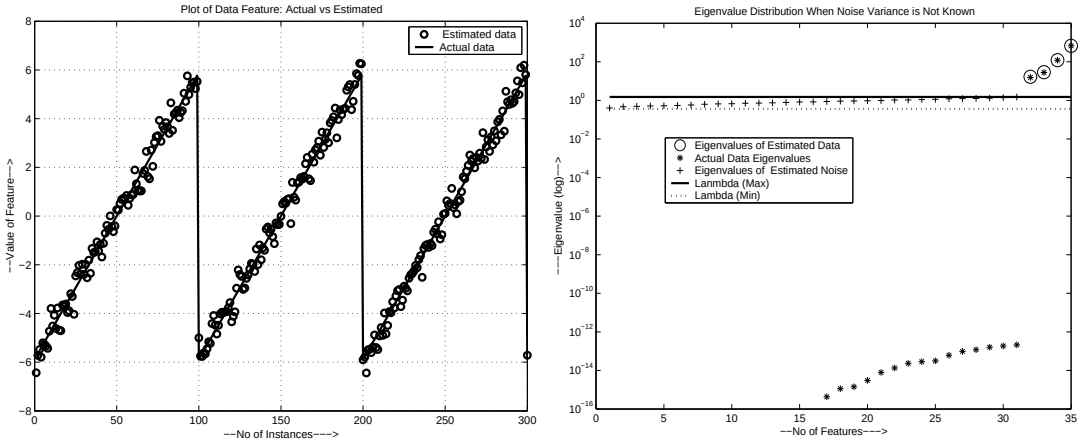

Figure 14: (Left) Estimation of actual data when the noise distribution is not known. (Right) Distribution of Eigenvalues of actual data, estimated data and estimated noise when noise variance is not known.

independent then there will be no correlation between the virtual attributes associated with the different columns. Clearly, the technique does not work when the features are independent of each other. However, a univariate data set with independent observations defines only a small fraction of the common data mining applications. We are not claiming that the technique will work for every type of data set. However, even if it works for certain types then it is a problem for the privacy-preserving technique.

The performance of the filtering algorithm also depends on the signal-to-noise ratio (SNR). The method seems to work reasonably well with a good deal of relative noise added for preserving privacy. Our experiments show that as long as the SNR is greater than 1 (i.e relative noise content does not exceed 100% of actual data value), the method works well for the data sets considered in this paper.

The original paper [2] on random-value-perturbation-based privacy-preserving data mining does not specify the specific SNR values for the experiments. However, all the data values they used appear to be within -0.5 to +1.5. They added Gaussian noise with variance 0.0625. Although they did not report the signal-to-noise ratio, we estimated the average SNR value to be approximately 2.2. Clearly this is greater than an SNR value of 1 and therefore the level of noise that they used for protecting privacy is well within the reach of the spectral filtering technique presented here.

The spectral filtering method does not distinguish between random noise component that may be inherent to the original data and noise explicitly added to the data for preserving data privacy. Therefore, it will

filter out both the artificially added and the inherent random noise components. Therefore, in some cases the estimated data may look different from the original data since the inherent noise component is taken out. However, one can argue that the inherent random component of the data may have little value as far as data mining is concerned. After all, data miners apply many data pre-processing techniques just for filtering out the underlying inherent noise in the data. Since the filtered data contains all the non-random patterns that may actually turn out to be the most relevant information as far as privacy preservation and data mining are concerned. The utility of the filtered inherent noise component fundamentally depends on the definition of privacy in the given application.

The analysis presented in Section VI makes several assumptions. We list those explicitly in the following for further clarifying the technique:

1. The original data U and the perturbation V are statistically independent (actually a weaker uncorrelatedness assumption would suffice). This is same as the assumptions made in most random additive noise-based techniques for privacy-preserving data mining. Use of correlated noise is also likely to make the pattern detection (e.g. distribution estimation, association rules) process more difficult. Our conjecture is that there may be a "no free lunch"-style theorem at large regarding this.
2. The eigenvectors of the covariance matrix $U_p^T U_p$ of the perturbed data are orthogonal to the eigenvectors of the covariance matrix $U^T U$ of the original data. We have addressed this in our discussion following the proof of Lemma 1. Matrix perturbation theory provides bounds on the angle between the eigenvectors of a matrix and that of a perturbation of the same matrix. These bounds can be used to quantify the orthogonality in terms of the norm of the perturbation matrix $V^T V$. These bounds have also been provided in our discussion following Lemma 1. We have determined this assumption to hold true in the experiments reported in this paper.
3. We assume that the asymptotic expressions for λ_{min} and λ_{max} can be used for sufficiently large, but finite, sample sizes. This is true for typical data sizes (both number of features and number of observations) encountered in data mining. Indeed, one of the useful features of the random matrix theory is that the asymptotics are derived in the limit as both the rows M (no. of observations) and columns N (no. of features) go to infinity, and for any ratio $Q = M/N$. Therefore, the asymptotic expressions are valid for a value of M that is comparable to that of N. In other words, we do not require the number of observations to go to infinity for a fixed number of features. A relatively small number of observations (as a function of the number of features) is usually sufficient to get good estimates. This is true for most realistic data sets encountered in data mining.
4. The range of eigenvalues of the original data (signal) matrix and those of noise do not overlap. This assumption is also true for most real-life data sets, where the signal matrix in the eigenvalue/eigenvector domain is usually characterized by a small number of dominant eigenvalues. Indeed, in many interesting applications, the top few eigenvalues account for more than 90% of the total signal energy (that is why Principal Component Analysis is so effective and popular!). It is true that there certainly may exist examples of signals (real-life or contrived) for which a significant portion of its eigenvalues lie in the interval $[\lambda_{min}, \lambda_{max}]$ of the eigenvalues of a random matrix. However, that does not invalidate the main claim of our paper that perturbation by additive noise may not preserve a whole lot of privacy in many cases. If a privacy-preserving technique does not work well for many data sets then we must recognize that and look for better, more secured solutions.

The paper offers a random-matrix based data filtering technique that may find wider application in developing a new perspective toward developing better privacy-preserving data mining algorithms. For example, we may be able to use this framework to develop algorithms that explicitly guard against potential compromise on privacy through linear transformations. The current privacy-preserving data mining algorithms do not pay adequate attention to this issue.

The filtering technique itself may find various applications in data mining. This random matrix based approach to separating the information bearing and noisy eigen-states also has potential computational advantages. Indeed, since the upper bound $\lambda_{\max}$ of the noisy eigenvalues is known a priori, one can easily use a suitable numerical technique (e.g., power method [11]) to compute just the few largest eigenvalues. Once these eigenvalues and corresponding eigenvectors are computed, one can obtain the actual-data-part of the covariance matrix, which can be subtracted off from the total covariance to isolate the noise-part of the covariance.

Since the problem mainly originates from the usage of additive, independent "white" noise for privacy preservation, we should explore "colored" noise for this application. We have already started exploring multiplicative noise matrices in this context. If U be the data matrix and V be an appropriately sized random noise matrix then we are interested in the properties of the perturbed data $U_p = UV$ for privacy-preserving data mining applications. If V is a square matrix then we may be able to extract signal using techniques like independent component analysis [10]. However, projection matrices that satisfy certain conditions may be more appealing for such applications. More details about this possibility can be found elsewhere [20].

Acknowledgments

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

References

[1] D. Agrawal and C. C. Aggawal. On the design and quantification of privacy preserving data mining algorothms. In *Proceedings of the 20th ACM SIMOD Symposium on Principles of Database Systems*, pages 247–255, Santa Barbara, May 2001.

[2] R. Agrawal and R. Srikant. Privacy-preserving data mining. In *Proceeding of the ACM SIGMOD Conference on Management of Data*, pages 439–450, Dallas, Texas, May 2000. ACM Press.

[3] Z. D. Bai, J. W. Silverstein, and Y. Q. Yin. A note on the largest eigenvalue of a large dimensional sample covariance matrix. *Journal of Multivariate Analysis*, 26(2):166–168, August 1988.

[4] W. Du and M. J. Atallah. Secure multi-party computation problems and their applications: A review and open problems. In *New Security Paradigms Workshop*, pages 11 – 20, 2001.

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

[6] A. Evfimevski, R. Srikant, R. Agrawal, and J. Gehrke. Privacy preserving mining of association rules. In *Proceedings of the ACM SIKDD Conference*, Edmonton, Canada, 2002.

[7] S. Evfimievski. Randomization techniques for privacy preserving association rule mining. In *SIGKDD Explorations*, volume 4(2), Dec 2002.

[8] S. Geman. A limit theorem for the norm of random matrices. *The Annals of Probability*, 8(2):252–261, April 1980.

[9] U. Grenander and J. W. Silverstein. Spectral analysis of networks with random topologies. *SIAM Journal on Applied Mathematics*, 32(2):499–519, 1977.

[10] F. Ham, N. Faour, and J. Wheeler. Infrasound signal separation using independent component analysis. In *21st Seismic Reearch Symposium*, 1999.

[11] J. E. Jackson. *A User's Guide to Principal Components*. John Wiley, 1991.

[12] S. Janson, T. L. , and A. Rucinski. *Random Graphs*. Wiley Publishers, 1 edition, 2000.

[13] E. Johnson and H. Kargupta. Collective, hierarchical clustering from distributed, heterogeneous data. In *Lecture Notes in Computer Science, volume 1759*, pages 221–244, 1999.

[14] D. Jonsson. Some limit theorems for the eigenvalues of a sample covariance matrix. *Journal of Multivariate Analysis*, 12:1–38, 1982.

[15] M. Kantarcioglu and C. Clifton. Privacy-preserving distributed mining of association rules on horizontally partitioned data. In *SIGMOD Workshop on DMKD*, Madison, WI, June 2002.

[16] H. Kargupta, B. Park, D. Hershberger, and E. Johnson. Collective data mining: a new perspective towards distributed data mining. In *Advances in Distributed and Parallel Knowledge Discovery, Eds: Kargupta, Hillol and Chan, Philip*. AAAI/MIT Press, 2000.

[17] H. Kargupta, H. Park, S. Pittie, L. Liu, D. Kushraj, and K. Sarkar. MobiMine: Monitoring the stock market from a PDA. *ACM SIGKDD Explorations*, 3:37–47, 2001.

[18] H. Kargupta, K. Sivakumar, and S. Ghosh. Dependency detection in mobimine and random matrices. In *Proceedings of the 6th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 250–262. Springer, 2002.

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

[20] K. Liu, H. Kargupta, and J. Ryan. Random projection and privacy preserving correlation computation from distributed data. Technical report, University of Maryland Baltimore County, Computer Science and Electrical Engineering Department, Technical Report TR-CS-03-24, 2003.

[21] D. G. Manolakis, V. K. Ingle, and S. M. Kogon. *Statistical and Adaptive Signal Processing*. McGraw Hill, 2000.

[22] V. A. Marcenko and L. A. Pastur. Distribution of eigenvalues for some sets of random matrices. *Mathematics of the USSR — Sbornik*, 1(4):457–483, 1967.

[23] M. L. Mehta. *Random Matrices*. Academic Press, London, 2 edition, 1991.

[24] A. Papoulis and S. U. Pillai. *Probability, Random Variables and Stochastic Processes*. McGraw Hill, New York, fourth edition, 2002.

[25] B. Park, R. Ayyagari, and H. Kargupta. A fourier analysis-based approach to learn classifier from distributed heterogeneous data. In *Proceedings of the First SIAM Internation Conference on Data Mining*, Chicago, US, 2001.

[26] B. H. Park and H. Kargupta. Distributed data mining: Algorithms, systems, and applications. In *Data Mining Handbook*, To be published, 2002.

[27] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1):81–106, 1986.

[28] U. M. L. Repository. http://www.ics.uci.edu/ mlearn/mlsummary.html.

[29] S. J. Rizvi and J. R. Haritsa. Maintaining data privacy in association rule mining. In *Proceedings of the 28th VLDB Conference*, Hong Kong, China, 2002.

[30] B. Schneier. Applied cryptography. John Wiley and Sons, 1995.

[31] J. W. Silverstein. On the weak limit of the largest eigenvalue of a large dimensional sample covariance matrix. *Journal of Multivariate Analysis*, 30(2):307–311, August 1989.

[32] J. W. Silverstein and P. L. Combettes. Signal detection via spectral theory of large dimensional random matrices. *IEEE Transactions on Signal Processing*, 40(8):2100–2105, 1992.

[33] G. W. Stewart. Error and perturbation bounds for subspaces associated with certain eigenvalue problems. *SIAM Review*, 15(4):727–764, October 1973.

[34] S. Stolfo et al. Jam: Java agents for meta-learning over distributed databases. In *Proceedings Third International Conference on Knowledge Discovery and Data Mining*, pages 74–81, Menlo Park, CA, 1997. AAAI Press.

[35] J. F. Traub, Y. Yemini, and H. Woz'niakowski. The statistical security of a statistical database. *ACM Transactions on Database Systems (TODS)*, 9(4):672–679, 1984.

[36] J. Vaidya and C. Clifton. Privacy preserving association rule mining in vertically partitioned data. In *The Eighth ACM SIGKDD International conference on Knowledge Discovery and Data Mining*, Edmonton, Alberta, CA, July 2002.

[37] H. Weyl. Inequalities between the two kinds of eigenvalues of a linear transformation. In *Proceedings of the National Academy of Sciences*, volume 35, pages 408–411, 1949.

[38] E. P. Wigner. On the statistical distribution of the widths and spacings of nuclear resonance levels. *Proceedings of the Cambridge Philosophical Society*, 47:790–798, 1952.

[39] Y. Q. Yin, Z. D. Bai, and P. R. Krishnaiah. On the limit of the largest eigenvalue of the large dimensional sample covariance matrix. *Probability Theory and Related Fields*, 78(4):509–521, August 1988.