A Random Rotation Perturbation Approach to Privacy Preserving Data Classification

Keke Chen Ling Liu
College of Computing, Georgia Institute of Technology
{kekechen, lingliu}@cc.gatech.edu

Abstract

This paper presents a random rotation perturbation approach for privacy preserving data classification. Concretely, we identify the importance of classification-specific information with respect to the loss of information factor, and present a random rotation perturbation framework for privacy preserving data classification. Our approach has two unique characteristics. First, we identify that many classification models utilize the geometric properties of datasets, which can be preserved by geometric rotation. We prove that the three types of classifiers will deliver the same performance over the rotation perturbed dataset as over the original dataset. Second, we propose a multi-column privacy model to address the problems of evaluating privacy quality for multidimensional perturbation. With this metric, we develop a local optimal algorithm to find the good rotation perturbation in terms of privacy guarantee. We also analyze both naive estimation and ICA-based reconstruction attacks with the privacy model. Our initial experiments show that the random rotation approach can provide high privacy guarantee while maintaining zero-loss of accuracy for the discussed classifiers.

1 Introduction

We are entering a highly connected information-intensive era. This information age has enabled organizations to collect large amount of data continuously. Many organizations wish to discover and study interesting patterns and trends over the large collections of datasets to improve their productivity and competitiveness. Privacy preserving data mining has become an important enabling technology for integrating data and mining interesting patterns from private collections of databases. This has resulted in a considerable amount of work on privacy preserving data mining methods in recent years such as [1, 3, 5, 2, 8, 9, 15, 18, 19], etc.

Data perturbation techniques are one of the most popular models for privacy preserving data mining [3, 1]. It is especially convenient for applications where the data owners need to export/publish the privacy-sensitive data. A data perturbation procedure can be simply described as follows. Before the data owner publishes the data, they *randomly* change the data in certain way to disguise the sensitive information while preserving the particular data property that is critical for building meaningful data mining models. Several perturbation techniques have been proposed recently, among which the most popular ones are randomization approach [3] and condensation approach [1]. In this paper, we will propose a new data perturbation technique specifically for a class of popular data classification mining models.

Loss of Privacy vs. Loss of Information.
Perturbation techniques are often evaluated with two basic metrics, loss of privacy and loss of model-specific information (resulting in loss of accuracy for data classification). An ideal data perturbation algorithm aims at minimizing both privacy loss and information loss. However, the two metrics are not well-balanced in many existing perturbation techniques [3, 2, 7, 1].

Loss of privacy can be intuitively described as the difficulty level in estimating the original values from the perturbed data. The more difficult the original values are estimated, the less loss of privacy is. In [3], the variance of the added random noise is used as the level of difficulty for estimating the original values. However, later research [7, 2] reveals that variance is not an effective indicator for random noise addition since the original data distribution is known − if a particular data distribution is considered, certain part of data in the distribution cannot be effectively protected. In addition, [14] shows that the loss of privacy is also subject to the special attacks that can reconstruct the original data from the perturbed data.

Loss of information typically refers to the amount of critical information preserved about the data sets after the perturbation. However, different data mining tasks, such as classification mining and association rule mining, typically utilize different set of information about the data sets. Existing techniques do not explicitly address that the critical information is actually task-specific. We argue that the information to be preserved after data perturbation should be highly specific to the mining tasks and even to a particular model. For example, the task

of building decision trees primarily concerns the column distribution. Hence, the quality of preserving column distribution becomes the key in applying randomization approach [3] to decision tree model. In comparison, the K-Nearest-Neighbor (KNN) model concerns primarily the distance relationship, nothing to do with the column distribution. We observed that, most classification models like KNN typically concern the multi-dimensional information rather than single column distribution. Thus, the perturbation is required to preserve multi-dimensional task-specific information rather than single dimensional information. To our knowledge, very few perturbation-based privacy protection proposals so far have considered *multi-dimensional perturbation techniques*.

Interesting to note is that the loss of privacy metric and the loss of information metric have exhibited contradictory rather than complimentary results in existing data perturbation techniques [3, 2, 7, 1]. Typically data perturbation algorithms that aims at minimizing the loss of privacy often have to bear with higher information loss. The intrinsic correlation between the loss of privacy and the loss of information raises a number of important issues regarding how to find a right balance between the two measures and how to build a data perturbation algorithm that ensures desired privacy requirements and yet minimizes the loss of information for the specific data mining task.

Contribution and Scope of the paper.

Bearing these issues in mind, we have developed a random rotation perturbation approach to privacy preserving data classification. In contrast to other existing privacy preserving classification methods [1, 3, 9, 15], our random rotation based perturbation exploits the task-specific multi-dimensional information about the datasets to be classified, which is critical to a large category of classification algorithms, and aims at producing a robust data perturbation that exhibits a better balance between loss of privacy and loss of information.

Concretely, we observe that the multi-dimensional geometric properties of datasets are the critical "task-specific information" for many classification algorithms. By preserving multi-dimensional geometric properties of the original dataset, classifiers trained over the perturbed dataset presents the same quality as classifiers over the original dataset. One intuitive way to preserve the multi-dimensional geometric properties is to perturb the original dataset through geometric rotation transformation. We have identified and proved that kernel methods, SVM classifiers with the three popular kernels, and the hyperplane-based classifiers, are the three categories of classifiers that are "rotation-invariant".

Another important challenge for the random rotation perturbation approach is the privacy loss measurement (the level of uncertainty) and privacy assurance (the resilience of the rotation transformation against unauthorized disclosure). Given that a random rotation based perturbation is a multi-dimensional perturbation, the privacy guarantee of the multiple dimensions (attributes) should be evaluated collectively to ensure the privacy of all columns involved and the privacy of the multi-column correlations. We design a unified privacy model to tackle the problem of privacy evaluation for multi-dimensional perturbation, which addresses three types of possible attacks: direct estimation, approximate reconstruction, and distribution-based inference attacks.

With the unified privacy metric, we present the privacy assurance of the random rotation perturbation as an optimization problem: given that all rotation transformations result in zero-loss of accuracy for the discussed classifiers, we want to pick one rotation matrix that provides higher privacy guarantee and stronger resilience against the inference attacks. Our experiments demonstrate that with our attack resilient random rotation selection algorithm, our random rotation perturbation can achieve much higher privacy guarantee and more robust in countering inference attacks than other existing perturbation techniques.

In a nutshell, random rotation perturbation refines the definition of loss of privacy and loss of information for multidimensional perturbation, and provides a particular method for "conveniently raising the privacy guarantee without loss of accuracy for the data classification task".

The rest of paper is organized as follows. Section 2 briefly reviews the related work. In Section 3, we describe the properties of geometric rotation transformation and prove that the three most popular categories of classifiers are invariant to rotation. Properties of general linear transformation are also briefly discussed. Section 4 introduces a general-purpose privacy measurement model for multi-column data perturbation and characterizes the privacy property of the rotation-based perturbation in terms of this metric. Three types of inference attacks are analyzed under this privacy model. We present the experimental results in Section 5 and conclude our work in section 6.

2 Related Work

A considerable amount of work on privacy preserving data mining methods have been reported in recent years [1, 3, 5, 2, 8, 19], etc. The most relevant work about perturbation techniques includes the random noise addition methods and the condensation-based perturbation technique. We below focus our discussion on these two sets of techniques and discuss their weakness in the context of privacy preserving data classification.

Random Noise Addition Approach

The random noise addition approach can be briefly described as follows. Suppose that the original values $(x_1, x_2, \ldots, x_n)$ from a column are randomly drawn from a random variable $\mathbf{X}$, which has some kind of distribution. The randomization process changes the original data with $\mathbf{Y} = \mathbf{X} + \mathbf{R}$, where $\mathbf{R}$ is a zero mean random noise. The resulting tuples $(x_1 + r_1, x_2 + r_2, \ldots, x_n + r_n)$ and the distribution of $\mathbf{R}$ are published. A reconstruction al-

gorithm is developed in [3] to construct the distribution of $\mathbf{X}$ based on the perturbed data and the distribution of $\mathbf{R}$. In particular, an expectation-maximization (EM) reconstruction algorithm was proposed in [2]. The distribution reconstructed by EM algorithm is proved to converge to the maximum likelihood estimate of the original distribution. A new decision-tree algorithm for the randomization approach is developed in [3], in order to build the decision tree from the perturbed data. Randomization approach is also used in privacy-preserving association-rule mining [8].

While the randomization approach is intuitive, several researchers have recently identified privacy breaches as one of the major problems with the randomization approach. Kargupta et al. [14, 11] observed that the spectral properties of the randomized data can be utilized to separate noise from the private data. The filtering algorithms based on random matrix theory are used to approximately reconstruct the private data from the perturbed data. The authors demonstrated that the randomization approach preserves little privacy in many cases.

Furthermore, there has been research [1] addressing other weaknesses associated with the value based randomization approach. For example, most of existing randomization and distribution reconstruction algorithms only concern about preserving the distribution of single columns. There has been surprisingly little attention paid on preserving value distributions over multiple correlated dimensions. Second, value-based randomization approach needs to develop new distribution-based classification algorithms. In contrast, our random rotation perturbation approach does not require modify existing data classification algorithms when applied to perturbed datasets. This is a clear advantage over techniques such as the method discussed in [3].

The randomization approach is also generalized by [7] and [4]. [7] proposes a refined privacy metric for the general randomization approach, and [4] develops a framework based on the refined privacy metric to improve the balance between the privacy and accuracy.

Condensation-based perturbation approach The condensation approach [1] aims at preserving the covariance matrix for multiple columns. Different from the randomization approach, it perturbs multiple columns as a whole to generate entire "perturbed dataset". The authors argue that the perturbed dataset preserves the covariance matrix, and thus, most existing data mining algorithms can be applied directly to the perturbed dataset without redeveloping any new algorithms.

The condensation approach can be briefly described as follows. It starts by partitioning the original data into k-record groups. Each group is formed by two steps – randomly select a record from the existing records as the center of group, and then find the $(k-1)$ nearest neighbors of the center as the other $(k-1)$ members. The selected k records are removed from the original dataset before forming the next group. Since each group has small locality, it is possible to regenerate a set of k records to approximately preserve the distribution and covariance. The record regeneration algorithm tries to preserve the eigenvectors and eigenvalues of each group. As a result, the distribution and the covariance of the points in the group are approximately preserved as shown in Figure 1. The authors demonstrated that the condensation approach can preserve data covariance well, and thus will not significantly sacrifice the accuracy of classifiers if the classifiers are trained with the perturbed data.

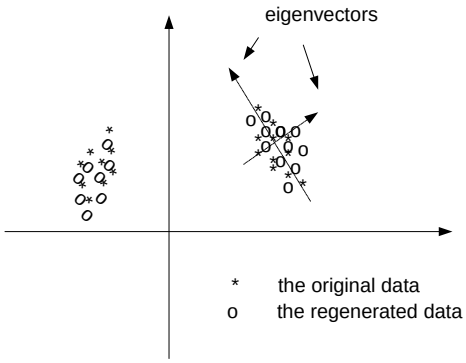

Figure 1. Condensation approach

However, we have observed that the condensation approach is weak in protecting the private data. The KNN-based data groups result in some serious conflicts between preserving covariance information and preserving privacy. As the authors claim, the smaller the size of the locality in each group, the better the quality of preserving the covariance with the regenerated k records is. Note that the regenerated k records are confined in the small spatial locality as Figure 1 shows. We design an algorithm that tries to find the nearest neighbor in the original data for each regenerated record. The result (section 5) shows that the difference between the regenerated records and the nearest neighbor in original data are very small, and thus, the original data records can be estimated from the perturbed data with high confidence.

3 Rotation Transformation and Data Classification

In this section, we first identify the set of geometric properties of the datasets, which are significant to most classification algorithms. Then we describe the definition of a rotation-based perturbation, and will discuss the effect of geometric transformations to three categories of popular classification algorithms. In particular, we will discuss the *rotation transformation*. Before entering concrete discussion, we define the notations for datasets.

Training Dataset and Unclassified Dataset. Training dataset is the part of data that has to be exported/published in privacy-preserving data classification. A classifier learns the classification model from the training data and

then is applied to classify the unclassified data. Suppose that X is a training dataset consisting of N data rows (records) and d columns (attributes). For the convenience of mathematical manipulation, we use $X_{d\times N}$ to notate the dataset, i.e., $X = [\mathbf{x}_1 \dots \mathbf{x}_N]$, where $\mathbf{x}_i$ is a data tuple, representing a vector in the real space $\mathbb{R}^d$. Each data tuple belongs to a predefined class, which is determined by its class label attribute y_i. The class labels can be nominal (or continuous for regression). The class label attribute of the data tuple is public, i.e., privacy-insensitive. All other attributes containing private information needs to be protected. Unclassified dataset could also be expored/published with privacy-protection if necessary.

3.1 Properties of Geometric Rotation

Let $R_{d\times d}$ represent the rotation matrix. Geometric rotation of the data X is generally notated as a function $g(X)$, $g(X) = RX$. Note that the transformation will not change the class label of data tuples, i.e., $R\mathbf{x}_i$, the rotation of data record $\mathbf{x}_i$, still has the label y_i.

A rotation matrix $R_{d\times d}$ is defined as a matrix having the follows properties. Let R^T represent the transpose of the matrix R, r_{ij} represent the (i, j) element of R, and I be the identity matrix. Both the rows and the columns of R are *orthonormal* [16], i.e., for any column j, $\sum_{i=1}^{d} r_{ij}^2 = 1$, and for any two columns j and k, $\sum_{i=1}^{d} r_{ij}r_{ik} = 0$. The similar property is held for rows. The definition infers that $R^TR = RR^T = I$. It also implies that by changing the order of the rows or columns of rotation matrix, the resulting matrix is still a rotation matrix. A random rotation matrix can be efficiently generated following the Haar distribution [17].

A key feature of rotation transformation is preserving length. Let $\mathbf{x}^T$ represent the transpose of vector $\mathbf{x}$, and $\| \mathbf{x} \| = \mathbf{x}^T\mathbf{x}$ represent the length of a vector $\mathbf{x}$. By the definition of rotation matrix, we have $\| R\mathbf{x} \| = \| \mathbf{x} \|$ Thus, rotation also preserves the Euclidean distance between any pair of points $\mathbf{x}$ and $\mathbf{y}$, due to $\| R(\mathbf{x} - \mathbf{y}) \| = \| \mathbf{x} - \mathbf{y} \|$.

Similarly, the inner product is also invariant to rotation. Let $< \mathbf{x}, \mathbf{y} > = \mathbf{x}^T\mathbf{y}$ represent the inner product of $\mathbf{x}$ and $\mathbf{y}$. We have $< R\mathbf{x}, R\mathbf{y} > = \mathbf{x}^T R^T R\mathbf{y} = < \mathbf{x}, \mathbf{y} >$.

Intuitively, rotation also preserves the geometric shapes such as hyperplane and hyper curved surface in the multi-dimensional space.

3.2 Rotation-invariant Classifiers

We first define the concept of "transformation-invariant classifiers", and then discuss the concrete classifiers having certain property. We say a classification algorithm is invariant to a transformation, if the classifier trained using the transformed data has the similar accuracy as that trained by the original data. We formally define a transformation-invariant classifier as follows.

We can treat the classification problem as function approximation problem – the classifiers are the functions learned from the training data [10]. Therefore, we can use functions to represent the classifiers. Let $\hat{f}_X$ represent a classifier $\hat{f}$ trained with dataset X and $\hat{f}_X(Y)$ be the classification result on dataset Y. Let $T(X)$ be any transformation function, which transforms the dataset X to another dataset X'. We use $Err(\hat{f}_X(Y))$ to notate the error rate of classifier $\hat{f}_X$ on testing data Y and let ε be some small real number, $|\varepsilon| < 1$.

Definition 1. *A classifier $\hat{f}$ is invariant to some transformation T if and only if $Err(\hat{f}_X(Y)) = Err(\hat{f}_{T(X)}(T(Y))) + \varepsilon$ for any training dataset X and testing dataset Y.*

With the strict condition $\hat{f}_X(Y) \equiv \hat{f}_{T(X)}(T(Y))$, we also have the following corollary.

Corollary 1. *In particular, if $\hat{f}_X(Y) \equiv \hat{f}_{T(X)}(T(Y))$, for any training dataset X and testing dataset Y, the classifier is invariant to the transformation $T(X)$.*

If a classifier $\hat{f}$ is invariant to *rotation* transformation, we specifically name it as a *rotation-invariant classifier*.

In the subsequent sections, we will prove that kernel methods, SVM classifiers with certain kernels, and hyperplane-based classifiers, are the three categories of classifiers that are rotation-invariant. The proofs are based on the strict condition given by Corollary 1.

KNN Classifiers and Kernel Methods

A KNN classifier determines the class label of a point by looking at the labels of its k nearest neighbors in the training dataset and classifies the point to the class that most of its neighbors belong to. Since the distances between any points are not changed after rotation, the k nearest neighbors are not changed and thus the classification result is not changed after rotation. Therefore, we have the first conclusion about the k Nearest Neighbor (KNN) classifiers.

Lemma 1. *KNN classifiers are rotation-invariant.*

KNN classifier is a special case of kernel methods. We assert that any kernel methods will be invariant to rotation too. Same as the KNN classifier, a traditional kernel method is a local classification method, which classifies the new data only based on the information from the neighbors in the training data.

Theorem 1. *Any kernel methods are invariant to rotation.*

Proof. Let us formally define the kernel methods first. In general, a kernel method also estimates the class label of a point $\mathbf{x}$ with the class labels of its neighbors. Let $K_\lambda(\mathbf{x}, \mathbf{x}_i)$ represent the weighting function of any point $\mathbf{x}_i$ in $\mathbf{x}$'s neighborhood, which is named as *kernel*. Let $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ be the points in the neighborhood of $\mathbf{x}$.

A kernel classifier for continuous class labels[1] is defined as,

$$\hat{f}_X(\mathbf{x}) = \frac{\sum_{i=1}^{n} K_\lambda(\mathbf{x}, \mathbf{x}_i) y_i}{\sum_{i=1}^{n} K_\lambda(\mathbf{x}, \mathbf{x}_i)} \quad (1)$$

Let λ be the width that determines the geometric area of the neighborhood at $\mathbf{x}$ [10]. The kernel $K_\lambda(\mathbf{x}, \mathbf{x}_i)$ is defined as,

$$K_\lambda(\mathbf{x}, \mathbf{x}_i) = D(\frac{\parallel \mathbf{x} - \mathbf{x}_i \parallel}{\lambda}) \quad (2)$$

$D(t)$ is a function, for example, $D(t) = \frac{1}{\sqrt{2\pi}} \exp\{-t^2/2\}$. Since $\parallel R\mathbf{x} - R\mathbf{x}_i \parallel = \parallel \mathbf{x} - \mathbf{x}_i \parallel$ and λ is constant, $D(t)$ is not changed after rotation and, thus, $K_\lambda(R\mathbf{x}, R\mathbf{x}_i) = K_\lambda(\mathbf{x}, \mathbf{x}_i)$. Since the geometric area around the point is not changed, the point set in the neighborhood of $R\mathbf{x}$ are still the rotation of those in the neighborhood of $\mathbf{x}$, i.e. $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\} \Rightarrow \{R\mathbf{x}_1, R\mathbf{x}_2, \ldots, R\mathbf{x}_n\}$ and these n points are used in $\hat{f}_{RX}$, which makes $\hat{f}_{RX}(R\mathbf{x}) = \hat{f}_X(\mathbf{x})$. □

Support Vector Machines

Support Vector Machine (SVM) classifiers also utilize kernel functions in training and classification. However, it uses the information from *all* points in the training set. Let y_i be the class label to a tuple $\mathbf{x}_i$ in the training set, α_i and β_0 be the parameters determined by training. A SVM classifier calculates the classification result of $\mathbf{x}$ using the following function.

$$\hat{f}_X(\mathbf{x}) = \sum_{i=1}^{N} \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + \beta_0 \quad (3)$$

Different from the kernel methods, which do not have a training procedure, we shall prove that SVM classifiers are invariant to rotation in two steps, 1) training with the rotated data results in the same set of parameters α_i and β_0; and 2) the classification function $\hat{f}$ is invariant to rotation.

Theorem 2. *SVM classifiers using polynomial, radial basis, and neural network kernels are invariant to rotation.*

Proof. The training problem is an optimization problem, which maximizes the Lagrangian (Wolfe) dual objective function [10]

$$L_D = \sum_{i=1}^{N} \alpha_i - 1/2 \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j)$$

subject to:

$$0 < \alpha_i < \gamma, \qquad \sum_{i=1}^{N} \alpha_i y_i = 0$$

[1] It has different form for discrete class labels, but the proof will be similar.

, where γ is a parameter chosen by the user, a larger γ corresponding to assigning a higher penalty to errors. We see that the training result of α_i is determined by the form of kernel function $K(\mathbf{x}_i, \mathbf{x}_j)$. Given α_i, β_0 can be determined by solving $y_i \hat{f}_X(\mathbf{x}_i) = 1$ for any $\mathbf{x}_i$ [10], which is again determined by the kernel function. Therefore, it is clear that if $K(R\mathbf{x}, R\mathbf{x}_i) = K(\mathbf{x}, \mathbf{x}_i)$ is held, the training procedure results in the same set of parameters.

There are the three popular choices for kernels listed in the SVM literature [6, 10].

d-th degree polynomial: $K(\mathbf{x}, \mathbf{x}') = (1 + < \mathbf{x}, \mathbf{x}' >)^d$,

radial basis: $K(\mathbf{x}, \mathbf{x}') = \exp(- \parallel \mathbf{x} - \mathbf{x}' \parallel / c)$,

neural network: $K(\mathbf{x}, \mathbf{x}') = \tanh(\kappa_1 < \mathbf{x}, \mathbf{x}' > + \kappa_2)$

Note that the three kernels only involve distance and inner product calculation. As we discussed in section 3.1, the two operations keep invariant to the rotation transformation. Apparently, $K(R\mathbf{x}, R\mathbf{x}') = K(\mathbf{x}, \mathbf{x}')$ are held for the three kernels. Therefore, training with the rotated data will not change the parameters for the SVM classifiers using the three popular kernels.

Similarly, $\hat{f}_X(\mathbf{x}) = \hat{f}_{RX}(R\mathbf{x})$ is held for the classification function (3) for the same reason. □

Perceptrons

Perceptron is the simplest neural network, which is a linear method for classification. We use perceptron as the representative example for hyperplane-based linear classifiers. The result for perceptron classifier can be easily generalized to all hyperplane-based linear classifiers.

A perceptron classifier uses a hyperplane to separate the training data, with the weights $\mathbf{w}^T = [w_1, \ldots, w_d]$ and bias β_0. The weights and bias parameters are determined by the training process. A trained classifier is represented as follows.

$$\hat{f}_X(\mathbf{x}) = \mathbf{w}^T \mathbf{x} + \beta_0$$

Theorem 3. *Perceptron classifiers are invariant to rotation.*

Proof. As Figure 2 shows, the hyperplane can be represented as $\mathbf{w}^T(\mathbf{x} - \mathbf{x_t}) = 0$, where $\mathbf{w}$ is the perpendicular axis to the hyperplane, and $\mathbf{x_t}$ represents the deviation of the plane from the origin (i.e., $\beta_0 = -\mathbf{w}^T\mathbf{x_t}$). Intuitively, rotation will make the classification hyperplane rotated as well, which rotates the perpendicular axis $\mathbf{w}$ to $R\mathbf{w}$ and the deviation $\mathbf{x}_t$ to $R\mathbf{x}_t$. Let $\mathbf{x}^r$ represent the data in the rotated space. The rotated hyperplane is represented as $(R\mathbf{w})^T(\mathbf{x}^r - R\mathbf{x}_t) = 0$, and the classifier is transformed to $\hat{f}_{RX}(\mathbf{x}^r) = \mathbf{w}^T R^T(\mathbf{x}^r - R\mathbf{x}_t)$. Since $\mathbf{x^r} = R\mathbf{x}$ and $R^T R = I$, $\hat{f}_{RX}(\mathbf{x}^r) = \mathbf{w}^T R^T R(\mathbf{x} - \mathbf{x_t}) = \mathbf{w}^T(\mathbf{x} - \mathbf{x_t})$ $= \hat{f}_X(\mathbf{x})$. The two classifiers are equivalent. □

In general, since rotation will preserve distance, density, and geometric shapes, any classifiers that find the decision boundary based on the geometric properties of the dataset, will still find the rotated decision boundary.

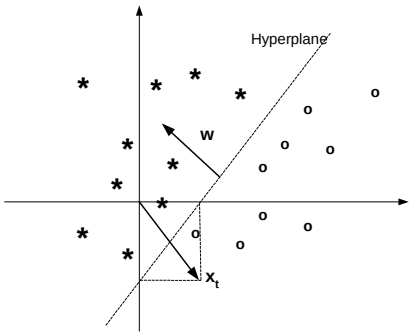

Figure 2. Hyperplane and its parameters

4 Evaluating Privacy Quality for Random Rotation Perturbation

The goals of rotation based data perturbation are twofold: preserving the accuracy of classifiers, and preserving the privacy of data. As we mentioned in the introduction, the loss of privacy and the loss of information (accuracy) are often considered as a pair of conflict factors for other existing data perturbation approaches. In contrast, a distinct feature of our rotation based perturbation approach is its clean separation of these two factors. The discussion about the rotation-invariant classifiers has proven that the rotation transformation theoretically guarantees zero-loss of accuracy for three popular types of classifiers, which makes the random rotation perturbation applicable to a large category of classification applications. We dedicate this section to discuss how good the rotation perturbation approach is in terms of preserving privacy.

The critical step to identify the *good* rotation perturbation is to define a multi-column privacy measure for evaluating the privacy quality of any rotation perturbation to a given dataset. With this privacy measure, we can employ some optimization methods to find the good rotation perturbations for a given dataset.

4.1 Privacy Model for Multi-column Perturbation

Unlike the existing value randomization methods, where multiple columns are perturbed separately, the random rotation perturbation needs to perturb *all* columns together. The privacy quality of all columns is correlated under one single transformation. Our approach to evaluating the privacy quality of random rotation perturbation consists of two steps: First, we define a general-purpose privacy metric that is effective for any multi-dimensional perturbation method. Then, the metric is applied to analyze the random rotation perturbation.

Since in practice different columns(attributes) may have different privacy concern, we consider that the general-purpose privacy metric Φ for entire dataset is based on **column privacy metric**. An abstract privacy model is defined as follows. Let $\mathbf{p}$ be the column privacy metric vector $\mathbf{p} = (p_1, p_2, \ldots, p_d)$, and there are **privacy weights** associated to the columns, respectively, notated as $\mathbf{w} = (w_1, w_2, \ldots, w_d)$. $\Phi = \Phi(\mathbf{p}, \mathbf{w})$ defines the privacy guarantee. Basically, the design of privacy model should consider determining the three factors $\mathbf{p}$, $\mathbf{w}$, and function Φ.

We will leave the concrete discussion about the design of $\mathbf{p}$ in the next section, and define the other two factors first. Since different columns may have different importance in terms of the level of privacy-sensitivity, the first design idea is to take the column importance into consideration. Let $\mathbf{w}$ denote the importance of columns in terms of preserving privacy. Intuitively, the more important the column is, the higher level of privacy guarantee will be required for the perturbed data, corresponding to that column. Therefore, we let $\sum_{i=1}^{d} w_i = 1$ and use p_i/w_i to represent the *weighted column privacy*.

The second intuition is the concept of *minimum privacy guarantee* among all columns. Concretely, when we measure the privacy quality of a multi-column perturbation, we need to pay special attention to the column having the lowest weighted column privacy, because such columns could become the breaking point of privacy. Hence, we design the first composition function $\Phi_1 = \min_{i=1}^{d}\{p_i/w_i\}$ and call it *minimum privacy guarantee*. Similarly, the *average privacy guarantee* of the multi-column perturbation $\Phi_2 = \frac{1}{d}\sum_{i=1}^{d} p_i/w_i$ is another interesting measure.

With the definition of privacy guarantee, we can evaluate the privacy quality of a give perturbation, and most importantly, we can use it to find the multi-dimensional perturbation that optimizes the privacy guarantee. With the rotation approach, we will demonstrate that it is convenient to adjust the perturbation method to considerably increase the privacy guarantee without compromising the accuracy of the classifiers.

4.2 Multi-column Privacy Analysis: A Unified Privacy Metric

Intuitively, for data perturbation approach, the quality of preserved privacy can be understood as the difficulty level of estimating the original data from the perturbed data. Basically, the attacks to the data perturbation techniques can be summarized in three categories: (1)estimating the original data directly from the perturbed data [3, 2], without any other knowledge about the data (naive inference); (2) approximately reconstructing the data from the perturbed data and then estimating the original data from the reconstructed data [14, 11] (approximation-based inference); and (3) if the distributions of the original columns are known, the values or the properties of the values in the particular part of the distribution can be estimated [2, 7] (distribution-based inference). A unified metric should be applicable to all three types of inference at-

tacks to determine the robustness of the perturbation technique. Due to the space limitation, we will not deal with the issues about distribution-oriented attacks to random rotation in this paper, and temporarily assume the column distributions are unknown to the users.

Let the difference between the original column data and the perturbed/reconstructed data be a random variable $\mathbf{D}$. Without any knowledge about the original data, the mean and variance of the difference present the level of difficulty for the estimation. Since the mean only presents the average difference, which is not a robust measure for protecting privacy, we choose to use the variance of the difference (VoD) as the primary metric to determine the level of difficulty in estimating the original data.

Let $\mathbf{Y}$ be a random variable, representing a column of the dataset, $\mathbf{Y}'$ be the perturbed/reconstructed result of $\mathbf{Y}$, and $\mathbf{D}$ be the difference between $\mathbf{Y}$ and $\mathbf{Y}'$. Thus we have $\mathbf{D} = \mathbf{Y}' - \mathbf{Y}$. Let $E[\mathbf{D}]$ and $Var(\mathbf{D})$ denote the mean and the variance of $\mathbf{D}$ respectively, y' be a perturbed/reconstructed value in $\mathbf{Y}'$, σ be the standard deviation of $\mathbf{D}$, and c denote some constant depending on the distribution of $\mathbf{D}$ and the confidence level. The corresponding original value y in $\mathbf{Y}$ is located in the range defined below:

$$[y' - E[\mathbf{D}] - c\sigma, y' - E[\mathbf{D}] + c\sigma]$$

The width of the estimation range, $2c\sigma$, presents the hardness to guess the original value (or amount of preserved privacy). In [3], $\mathbf{Y}'$ is defined as $\mathbf{Y}' = \mathbf{Y} + \mathbf{R}$, $\mathbf{R}$ represents a zero mean noise random variable. Therefore, $E[\mathbf{D}] = 0$ and the estimation solely depends on the distribution of the added random noise $\mathbf{R}$. For simplicity, we use σ to represent the privacy level.

To evaluate the privacy quality of multi-dimensional perturbation, we need to evaluate the privacy of all perturbed columns together. Unfortunately, the single-column privacy metric does not work across different columns since it ignores the effect of value range and the mean of the original data column. The same amount of VoD is not equally effective for different value ranges. One effective way to unify the different value ranges is via *normalization*. With normalization, the unified privacy metric is calculated in following three steps:

1. Let $s_i = 1/(max(\mathbf{Y}_i) - min(\mathbf{Y}_i))$, $t_i = min(\mathbf{Y}_i)/(max(\mathbf{Y}_i) - min(\mathbf{Y}_i))$ denote the constants that are determined by the value range of the column $\mathbf{Y}_i$. The column $\mathbf{Y}_i$ is scaled to range [0, 1], generating $\mathbf{Y}_{si}$, with the transformation $\mathbf{Y}_{si} = s_i(\mathbf{Y}_i - t_i)$. This allows all columns to be evaluated on the same base, eliminating the effect of diverse value ranges.

2. The normalized data $\mathbf{Y}_{si}$ is perturbed to $\mathbf{Y}'_{si}$. Let $\mathbf{D}'_i$ =$\mathbf{Y}'_{si} - \mathbf{Y}_{si}$. We use $Var(\mathbf{D}'_i)$, instead of $Var(\mathbf{D}_i)$, as the unified measure of privacy quality.

3. The unified column privacy metrics compose the privacy vector $\mathbf{p}$. The composition functions Φ_1 and Φ_2 are applied to calculate the minimum privacy guarantee and the average privacy guarantee, respectively.

This above evaluation should be applied to all of the three kinds of attacks and the lowest one should be considered as the final privacy guarantee.

4.3 Multi-column Privacy Analysis for Random Rotation Perturbation

With the variance metric over the normalized data, we can formally analyze the privacy quality of random rotation perturbation. Let X be the normalized dataset, X' be the rotation of X, and I_d be the d-dimensional identity matrix. Thus, VoD can be evaluated based on the difference matrix $X' - X$, and the VoD for i-th column is the element (i,i) in the covariance matrix of $X' - X$, which is represented as

$$\begin{aligned} & Cov(X' - X)_{(i,i)} = Cov(RX - X)_{(i,i)} \\ = \; & ((R - I_d)Cov(X)(R - I_d)^T)_{(i,i)} \end{aligned} \quad (4)$$

Let r_{ij} represent the element (i, j) in the matrix R, and c_{ij} be the element (i, j) in the covariance matrix of X. The VoD for ith column is computed as follows.

$$Cov(X' - X)_{(i,i)} = \sum_{j=1}^{d}\sum_{k=1}^{d} r_{ij}r_{ik}c_{kj} - 2\sum_{j=1}^{d} r_{ij}c_{ij} + c_{ii} \quad (5)$$

When the random rotation matrix generated following the Haar distribution, a considerable number of matrix entries are approximately independent normal N(0, 1/d) [13]. The full discussion about the numerical characteristics of the random rotation matrix is out of the scope of this paper. However, we can still get some observations from equation (5):

1. the mean level of VoD_i is affected by the variance of the original data column, i.e., c_{ii}. Large c_{ii} tends to give higher privacy level on average.

2. The variance of VoD_i affects the efficiency of randomization. The larger the $Var(VoD_i)$, the more likely the randomly generated rotation matrices can provide a high privacy level compared to the mean level of VoD_i. Exact form of $Var(VoD_i)$ should be complicated, but from the equation (5), we can see $Var(VoD_i)$ might be tightly related to the average of the squared covariance entries, i.e. $O(1/d^2 \sum_{i=1}^{d}\sum_{j=1}^{d} c_{ij})$.

3. VoD_i only considers the i-th row vectors of rotation matrix. Thus, it is possible to simply swap the rows of R to locally improve the overall privacy guarantee.

The third observation leads us to propose a row-swapping based fast local optimization method for finding a better rotation from a given rotation. This method can significantly reduce the search space and thus provides better efficiency. Our experimental result shows that, with the local optimization, the minimum privacy level can be increased by about 10% or more. We formalize the swapping-maximization method as follows: Consider a d-dimensional dataset. Let $\{(1), (2), \ldots, (d)\}$ be a permutation of the sequence $\{1, 2, \ldots, d\}$. Let the importance level of privacy preserving for the columns be $[w_1, w_2, \ldots, w_d]$. The goal is to find the permutation of rows that maximize the minimum or average privacy guarantee for a given rotation matrix.

$$argmax_{\{(1),(2),\ldots,(d)\}}\{ min_{1\leq i\leq d}\{(\sum_{j=1}^{d}\sum_{k=1}^{d} r_{(i)j}r_{(i)k}c_{kj} - 2\sum_{j=1}^{d} r_{(i)j}c_{ij} + c_{ii})/w_i\}\} \quad (6)$$

Since the matrix R' generated by swapping the rows of R is still a rotation matrix (recall section 3.1), the above local optimization step will not change the rotation-invariance property of the givenclassifiers.

The unified privacy metric evaluates the privacy guarantee and the resilience against nave inference – the first type of privacy attack. Considering the approximation-based inference – the second level of privacy attack through applying some reconstruction method to the random rotation perturbation, we identify that Independent Component Analysis (ICA) [12] could be applied to estimate the structure of the normalized dataset X. We dedicate the next section to analyze the ICA-based attacks and show that our rotation-based perturbation is robust to this type of inference attacks.

4.4 ICA-based Attack to Rotation Perturbation

Intuitively, one might think that the Independent Component Analysis (ICA) could be considered as the most commonly used method to breach the privacy protected by the random rotation perturbation approach. However, we argue that ICA is in general not effective in breaking the rotation perturbation in practice.

ICA is a fundamental problem in signal processing which is highly effective in several applications such as blind source separation [12] of mixed electroencephalographic(EEG) signals, audio signals and the analysis of functional magnetic resonance imaging (fMRI) data. Let matrix X composed by the source signals, where each row vector is a signal. Suppose we can observe the mixed signals X', which is generated by linear transformation $X' = AX$. ICA model can be applied to estimate the independent components (the row vectors) of the original signals X, from the mixed signals X', if the following conditions are satisfied:

1. The source signals are independent, i.e., the row vectors of X are independent;
2. All the source signals must be non-Gaussian with possible exception of one signal;
3. The number of observed signals, i.e. the number of row vectors of X', must be at least as large as the independent source signals.
4. The transformation matrix A must be of full column rank.

For rotation matrices, the 3rd and 4th conditions are always satisfied. However, the first two conditions, especially the independency condition, although practical for signal processing, seem not very common in data classification. In practice, the dependent source signals can be approximately regarded as one signal in ICA and people can often tolerate considerable errors in the applications of audio/video signal reconstruction, cracking the privacy of the original dataset X requires to exactly locate and precisely estimate the original row vectors. This has greatly restricted the effectiveness of ICA model based attacks to the rotation-based perturbation.

Concretely, there are two basic difficulties in applying the above ICA-based attack to the rotation-based perturbation. First of all, if there is significant dependency between any attributes, ICA fails to converge and results in less row vectors than the original ones, which cannot be used to effectively detect the private information. Second, even ICA can be done perfectly, the order of the original independent components cannot be preserved or determined through ICA [12]. Formally, any permutation matrix P and its inverse P^{-1} can be substituted in the model to give $X' = AP^{-1}PX$. ICA could possibly give the estimate for some permutated source PX. Thus, we cannot identify the particular column assuming that the original column distributions are unknown or perturbed.

The effectiveness of the ICA reconstruction method can be evaluated with the unified metric as well. The VoDs are now calculated based on the reconstructed data and the original data. Since the ordering of the reconstructed row vectors is not certain, we estimate the VoDs with the best effort – considering all of the $d!$ possible orderings and finding the most likely one. The most likely ordering is defined as the one that gives the lowest privacy guarantee among all of the orderings. Let $\hat{X}_k$ be the ICA reconstructed data $\hat{X}$ reordered with one of the row orderings, and p_k^{min} be the minimum privacy guarantee for $\hat{X}_k$, $k = 1 \ldots d!$, i.e., $p_k^{min} = min_{1\leq i\leq d}$ $\{\frac{1}{Nw_i}(Cov(\hat{X}_k - X)_{(i,i)}\}$. The ordering that gives lowest minimum privacy quality is selected as the most likely ordering.

We observed that, when there is certain dependency between the attributes (columns), the ICA method cannot effectively lower the privacy guarantee. More importantly, one can carefully select the rotation matrix such that the chosen perturbation is more resilient to the ICA-based attacks.

4.5 Selecting Rotation Center

Note that rotation does not perturb the points equally. The points near the rotation center will change less than those distant to the center. With the origin as the center, the small values close to 0 keep small after rotation, which is weak in protecting privacy. This can be remedied by randomly "floating" the rotation center so that the weakly perturbed points are not predictable. Concretely, the dimensional value of the center is uniformly drawn from the range [0, 1], so that the center is randomly selected in the normalized data space. The rotation transformation for non-origin centers is done by first translating the dataset to the center and then rotating the dataset. Let T be the translation matrix. The VoDs are not changed by translation due to the fact $Cov(R(X-T)-X) \equiv Cov(RX-X)$. When the center-translated rotation is applied to the original data, the center is simply scaled up (denormalized) by the parameters s_i and t_i defined earlier. Since translation preserves all of the basic geometric properties, the classifiers seeking the geometric decision boundary will be still invariant to translation.

4.6 Putting All Together: Randomized Algorithm for Finding a Better Rotation

We have discussed the unified privacy metric for evaluating the quality of a random rotation perturbation with the unified privacy metric. We have also shown how to choose the rotation matrix in order to maximize the unified metric in terms of the naive value estimation attack (naive inference) and reconstruction-based estimation attack (approximation-based inference). In addition, we choose to randomly optimize the rotation so that the attacker cannot inference anything from the optimization algorithm.

Algorithm 1 runs in a given number of iterations. Initially, the rotation center is randomly selected. In each iteration, the algorithm randomly generates a rotation matrix. Local maximization of variance through swapping rows is then applied to find a better rotation matrix, which is then tested by the ICA reconstruction. The rotation matrix is accepted as the currently best perturbation if it provides higher minimum privacy guarantee than the previous perturbations.

Algorithm 1 Finding a Better Rotation ($X_{d\times N}$, **w**, m)

Input: $X_{d\times N}$:the original dataset, **w**: weights of attributes in privacy evaluation, m: the number of iterations.
Output: R_t: the selected rotation matrix, T_r: the rotation center, p: privacy quality
calculate the covariance matrix C of X;
$p = 0$, and randomly generate the rotation center T_r;
for Each iteration **do**
  randomly generate a rotation matrix R;
  swapping the rows of R to get R', which maximizes $min_{1\le i\le d}\{\frac{1}{w_i}(Cov(R'X-X)_{(i,i)}\}$;
  $p_0 =$ the privacy quality of R', $p_1 = 0$;
  if $p_0 > p$ **then**
    generate $\hat{X}$ with ICA;
    $p_1 = \min\{p_k^{min}, k = 1\ldots d!\}$, $p_k^{min} = min_{1\le i\le d}\{\frac{1}{w_i}(Cov(\hat{X}_k - X)_{(i,i)}\}$;
  end if
  if $p < \min(p_0, p_1)$ **then**
    $p = \min(p_0, p_1)$, $R_t = R'$;
  end if
end for

5 Experimental Result

We design three sets of experiments. The first set is used to show that the discussed classifiers are invariant to rotations. The second set shows privacy quality of the good rotation perturbation. Finally, we compare the privacy quality between the condensation approach and the random rotation approach. All datasets used in the experiments can be found in UCI machine learning database [2].

5.1 Rotation-invariant Classifiers

In this experiment, we verify the invariance property of several classifiers discussed in section 3.2. Three classifiers: KNN classifier, SVM classifier with RBF kernel, and perceptron, are picked as the representative of the discussed three kinds of classifiers.

Each dataset is randomly rotated 10 times with different rotation matrices. Each of the 10 resultant datasets is used to train and cross-validate the classifiers. The reported numbers are the average of the 10 testing results. We calculate the difference of performance, i.e., accuracy, between the classifier trained with the original data and those trained with the rotated data.

In the table 1, 'orig' is the classifier accuracy to the original datasets, 'R' denotes the result of the classifiers trained with rotated data, and the numbers in 'R' columns are the performance difference between the classifiers trained with original and rotated data, for example, "-1.0 ± 0.2" means that the classifiers trained with the rotated data have the accuracy rate 1.0% lower than the original classifier on average, and the standard deviation is 0.2%. We use single-perceptron classifiers in the experiment. Therefore, the datasets having more than two classes, such as "E.Coli", "Iris" and "Wine" datasets, are not evaluated for perceptron classifier. It shows that the accuracy of the classifiers almost does not change when rotation is applied.

[2] http://www.ics.uci.edu/∼mlearn/Machine-Learning.html

Dataset	N	d	k	KNN		SVM(RBF)		Perceptron		LOP_{min}	LOP_{avg}	ICA_{min}	ICA_{avg}
				orig	R	orig	R	orig	R				
Breast-w	699	10	2	97.6	-0.5 ± 0.3	97.2	0 ± 0	89.1	-4.9 ± 1.2	0.41	0.50	0.73	0.95
Credit-a	690	14	2	82.7	$+0.2 \pm 0.8$	85.5	0 ± 0	64.6	$+4.7 \pm 1.5$	0.31	0.47	0.51*	0.97*
Credit-g	1000	24	2	72.1	$+1.2 \pm 0.9$	76.3	0 ± 0	70.1	-0.1 ± 0	0.40	0.51	0.52*	0.99*
Diabetes	768	8	2	73.3	$+0.4 \pm 0.5$	77.3	0 ± 0	66.6	-4.5 ± 0.8	0.23	0.28	0.81	0.95
E.Coli	336	7	8	85.1	$+0.2 \pm 0.8$	78.6	0 ± 0	-	-	0.24	0.34	0.75*	0.95*
Heart	270	13	2	78.9	$+2.1 \pm 0.5$	84.8	0 ± 0	67.4	-0.41 ± 1.0	0.42	0.54	0.50*	0.97*
Hepatitis	155	19	2	80.8	$+1.8 \pm 1.5$	79.4	0 ± 0	79.4	-0.3 ± 0.8	0.37	0.48	0.53	1.00
Ionosphere	351	34	2	86.4	$+0.5 \pm 0.6$	89.7	0 ± 0	66.9	-1.8 ± 0.6	0.31	0.41	0.82*	1.01*
Iris	150	4	3	94.6	$+1.2 \pm 0.4$	96.7	0 ± 0	-	-	0.43	0.50	0.69*	0.79*
Tic-tac-toe	958	9	2	99.0	-0.3 ± 0.4	98.3	0 ± 0	56.6	$+8.0 \pm 0.6$	0.61	0.68	0.52	0.88
Votes	435	16	2	92.5	$+0.4 \pm 0.4$	95.6	0 ± 0	60.3	-2.8 ± 1.3	0.65	0.82	0.50	0.99
Wine	178	13	3	98.3	-0.6 ± 0.5	98.9	0 ± 0	-	-	0.26	0.34	0.78*	0.97*

Table 1. Experimental result on transformation-invariant classifiers

5.2 Privacy Quality of Random Rotation Perturbation

We investigate the privacy property of the transformation approach with the multi-column privacy metric introduced in section 4. Each column is considered equally important in privacy preserving, thus, the weights are not included in evaluation. We use FastICA package, which can be downloaded from http://www.cis.hut.fi/projects/ica/fastica/, in evaluating the effectiveness of ICA-based reconstruction.

Right side of Table 1 summarizes the evaluation of privacy quality on the experimental datasets. The results are obtained in 50 iterations with Algorithm 1. The numbers are $\sqrt{VoD} = \sigma$, i.e., standard deviation of the difference between the normalized original data and the perturbed/reconstructed data (LOPs/ICAs). The column LOP_{min} represents the locally optimal minimum privacy guarantee in the 50 iterations. LOP_{avg} represents the locally optimal average privacy guarantee. ICA_{min} and ICA_{avg} represents the lowest minimum privacy and average privacy the ICA reconstruction can achieve in the 50 iterations, respectively. Among the 12 datasets, ICA does not converge for 7 datasets which are marked by '*' and thus not effectively reduce the privacy guarantee. For the rest 5 datasets, ICA can possibly reduce the privacy quality by some small amount, such as "Tic-tac-toe" and "Votes".

Figure 3 for dataset "Breast-Wisconsin" shows that data estimated by ineffective ICA reconstruction. In this case, the local optimized rotation perturbation is selected as the best perturbation. Figure 4 shows that ICA reconstruction may undermine the privacy quality for some datasets. In this case, the actual privacy guarantee will be located at between the locally optimized privacy guarantee and the ICA reconstruction lowered privacy guarantee, for we can always select a rotation matrix that is more resistent to ICA reconstruction. When it is detected that ICA reconstruction can seriously reduce the privacy guarantee, say, to less than 0.2, we need additional methods to perturb the data so that the conditions for effective ICA reconstruction are not satisfied. We leave this as a part of future work.

5.3 Rotation-based Approach vs. Condensation Approach.

We design a simple algorithm to estimate the privacy quality of condensation approach. As we mentioned, since the perturbation part is done within the KNN neighbors, it is highly possible that the perturbed data is in the KNN neighbors of the original data too. For each record in the perturbed dataset, we try to find the nearest neighbor in the original data. By comparing the difference between the perturbed data and its nearest neighbor in the original data, we can approximately measure the privacy quality of condensation approach.

Intuitively, the better locality the KNN perturbation is, the better the condensation approach can preserve the information, but the worse the privacy quality is. Figure 5 and 6 show the relationship between the size of condensation group and the privacy quality on "E.Coli" and "Diabetes" datasets. It was demonstrated in the paper [1] that the accuracy of classifiers becomes stable with the increase of the size of condensation group. However, we observed that the privacy quality generally stays low, no matter how the condensation size changes. Experiment on both datasets shows the minimum privacy guarantees are very low, neither are the average privacy levels. We also observed that the minimum privacy is 0 for "Ionosphere" data, which happens to contain one column that has the same value. Condensation method seems not working for such cases at all. Supported by the other two Figures (7 and 8), we can conclude that the condensation approach only provides weak privacy protection and we cannot possibly adjust the perturbation to meet the higher privacy requirement.

While the rotation approach provides almost zero-loss of information for classification, it also presents much higher privacy quality than the condensation approach. Figure 7 and 8 shows the comparison on the minimum privacy guarantee and the average privacy guarantee of the two approaches. The numbers for rotation approach are the results generated by the randomized algorithm in 50

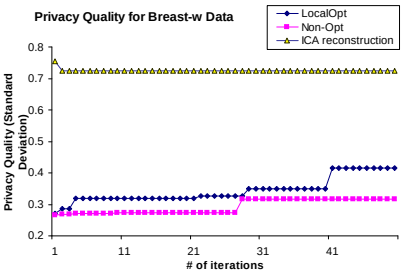

Figure 3. ICA reconstruction has no effect on privacy guarantee.

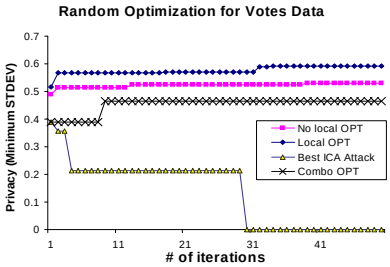

Figure 4. Example that ICA undermines the privacy guarantee.

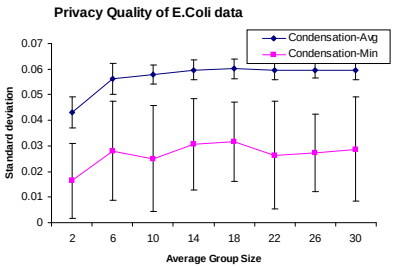

Figure 5. Privacy quality of condensation approach on E.Coli data.

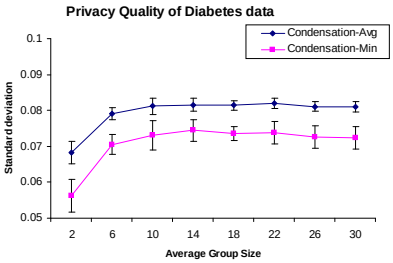

Figure 6. Privacy quality of condensation approach on Diabetes data.

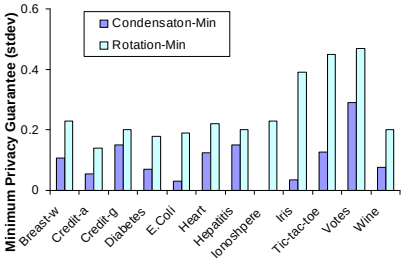

Figure 7. Comparison on minimum privacy level.

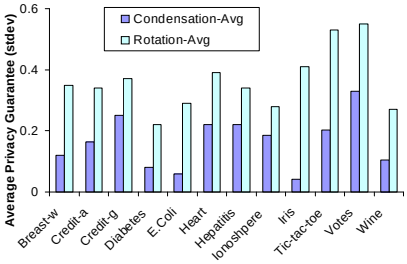

Figure 8. Comparison on average privacy level.

iterations. For exmaple, in Figure 7, "Rotation-Min" denotes the optimal minimum privacy guarantee, taking the ICA-attack into account as we discussed. We see that the rotation approach can easily provide much higher privacy level than the condensation approach.

6 Conclusion

We present a random rotation-based multidimensional perturbation approach for privacy preserving data classification. Geometric rotation can preserve the important geometric properties, thus most classifiers utilizing geometric class boundaries become invariant to the rotated data. We proved analytically and experimentally that the three popular types of classifiers (kernel methods, SVM classifiers with certain kernels, and hyperplane-based classifiers) are all invariant to rotation perturbation.

Random rotation perturbation perturbs multiple columns in one transformation, which introduces new challenges in evaluating the privacy guarantee for multidimensional perturbation. We design a unified privacy metric based on value-range normalization and multi-column privacy composition. With this unified privacy metric we are able to find the local optimal rotation perturbation in terms of privacy guarantee. The unified privacy metric also enables us to identify and analyze the resilience of the rotation perturbation approach against the ICA-based data reconstruction attacks. Our experimental result shows that the geometric rotation approach not only preserves the accuracy of the rotation-invariant classifiers, but also provides much higher privacy guarantee, compared to the existing multi-dimensional perturbation techniques.

References

[1] AGGARWAL, C. C., AND YU, P. S. A condensation approach to privacy preserving data mining. *Proc. of Intl. Conf. on Extending Database Technology (EDBT)* (2004).

[2] AGRAWAL, D., AND AGGARWAL, C. C. On the design and quantification of privacy preserving data mining algorithms. *Proc. of ACM PODS Conference* (2002).

[3] AGRAWAL, R., AND SRIKANT, R. Privacy-preserving data mining. *Proc. of ACM SIGMOD Conference* (2000).

[4] AGRAWAL, S., AND HARITSA, J. R. A framework for high-accuracy privacy-preserving mining. In *Proc. of IEEE Intl. Conf. on Data Eng. (ICDE)* (2005), pp. 193–204.

[5] CLIFTON, C. Tutorial: Privacy-preserving data mining. *Proc. of ACM SIGKDD Conference* (2003).

[6] CRISTIANINI, N., AND SHAWE-TAYLOR, J. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2000.

[7] EVFIMIEVSKI, A., GEHRKE, J., AND SRIKANT, R. Limiting privacy breaches in privacy preserving data mining. *Proc. of ACM PODS Conference* (2003).

[8] EVFIMIEVSKI, A., SRIKANT, R., AGRAWAL, R., AND GEHRKE, J. Privacy preserving mining of association rules. *Proc. of ACM SIGKDD Conference* (2002).

[9] FEIGENBAUM, J., ISHAI, Y., MALKIN, T., NISSIM, K., STRAUSS, M., AND WRIGHT, R. N. Secure multiparty computation of approximations. In *ICALP '01: Proceedings of the 28th International Colloquium on Automata, Languages and Programming*, (2001), Springer-Verlag, pp. 927–938.

[10] HASTIE, T., TIBSHIRANI, R., AND FRIEDMANN, J. *The Elements of Statistical Learning*. Springer-Verlag, 2001.

[11] HUANG, Z., DU, W., AND CHEN, B. Deriving private information from randomized data. *Proc. of ACM SIGMOD Conference* (2005).

[12] HYVARINEN, A., KARHUNEN, J., AND OJA, E. *Independent Component Analysis*. Wiley-Interscience, 2001.

[13] JIANG, T. How many entries in a typical orthogonal matrix can be approximated by independent normals. *To appear in The Annals of Probability* (2005).

[14] KARGUPTA, H., DATTA, S., WANG, Q., AND SIVAKUMAR, K. On the privacy preserving properties of random data perturbation techniques. *Proc. of Intl. Conf. on Data Mining (ICDM)* (2003).

[15] LINDELL, Y., AND PINKAS, B. Privacy preserving data mining. *Journal of Cryptology 15*, 3 (2000).

[16] SADUN, L. *Applied Linear Algebra: the Decoupling Principle*. Prentice Hall, 2001.

[17] STEWART, G. The efficient generation of random orthogonal matrices with an application to condition estimation. *SIAM Journal on Numerical Analysis 17* (1980).

[18] VAIDYA, J., AND CLIFTON, C. Privacy preserving association rule mining in vertically partitioned data. *Proc. of ACM SIGKDD Conference* (2002).

[19] VAIDYA, J., AND CLIFTON, C. Privacy preserving k-means clustering over vertically partitioned data. *Proc. of ACM SIGKDD Conference* (2003).