

Jie Wang*, Lian Liu, Dianwei Han, and Jun Zhang†
Department of Computer Science
University of Kentucky, Lexington, KY 40506-0046
jwanga@csr.uky.edu, jzhang@cs.uky.edu

September 13, 2007

Abstract

Data mining techniques enable discovery of valuable data patterns and knowledge in shared data and increase profitability and enhance national security. Security and privacy threats arising from the use of data mining techniques bring a risk of disclosure of confidential knowledge as data is made public. How to control the level of knowledge disclosure and secure certain confidential patterns is a subtask comparable to confidential data hiding in privacy preserving data mining. We propose a technique to simultaneously hide data values and confidential patterns without undesirable side effects on distorting nonconfidential patterns. We use nonnegative matrix factorization technique to distort the original dataset and preserve its overall characteristics. A factor swapping method is designed to hide particular confidential patterns in an unsupervised learning. The effectiveness of this novel hiding technique is examined by conducting k-means clustering on a benchmark dataset. Experimental results indicate that our technique can produce a single modified dataset to achieve both pattern and data value hiding. The usability of the data is well maintained. Under certain constraints on the nonnegative matrix factorization iterations, an optimal solution can be computed in which the user-specified confidential memberships or relationships are hidden without undesirable alterations on

nonconfidential patterns.

1 Introduction

Data mining techniques have now been used in commercial, industrial and governmental business, for various purposes, ranging from increasing profitability to enhancing national security. For example, interorganizational collaboration significantly improves supply chains and enables more rapid and less costly conduction of transactions among partners. Data sharing is a bridge of communications and collaboration. Data mining techniques can be broadly utilized to discover valuable knowledge in private or shared public data. In this situation, organizations and enterprises must fulfill two contradictory missions. One is to share data or information with other partners or the public. The other is to protect confidential data and privacy of the data subjects. This dilemma spurred a research area, known as privacy preserving data mining (PPDM). PPDM can be divided into two categories. One is data hiding to protect sensitive data values but maintain data patterns. Another is pattern hiding to protect sensitive patterns while keep usability of the original data.

A large amount of literatures in PPDM has fallen into the category of data hiding [3, 4, 7, 26]. Data hiding is used to prevent improper use of data. In many cases, control on the usage of data mining techniques should be considered. Pattern hiding, another security concern, grows out of the context of collaboration where sharing data is required among partners. For individual members in a collaborative project, preventing other partners from discovering some confidential knowledge is vital when competitors or partners can use data mining algorithms to ex-

*URL: <http://www.csr.uky.edu/~jwanga>

†The corresponding author. E-mail: jzhang@cs.uky.edu, URL: <http://www.cs.uky.edu/~jzhang>. This author's research work was supported in part by the U.S. National Science Foundation under grant CCF-0527967, in part by the National Institutes of Health under grant 1R01HL086644-01, in part by the Kentucky Science and Engineering Foundation under grant KSEF-148-502-06-186, and in part by the Alzheimer's Association under Grant NIGR-06-25460.

tract valuable (but potentially damaging) knowledge from the shared data. It was indicated as a threat to database security by O’Leary [23]. Clifton and Marks recently gave a well designed scenario [7].

Verykios *et al.* [27] analyzed a business situation to indicate the need to prevent not only the disclosure of confidential personal data, but also data mining techniques from discovering sensitive knowledge which may not even be known to the data owners. Clifton and Marks [7] propose some possible approaches to deal with such problems, including limiting access to the data, fuzzy-fying data, eliminating unnecessary groupings and augmenting the data. Compared to a rich literature on data hiding, the published research work on pattern hiding is mainly limited to association rule hiding and classification rule hiding [5, 8, 27, 30].

However, to the best of our knowledge, there has been no effort made on achieving data hiding and pattern hiding at the same time. Since data modification or reconstruction is utilized in most of the existing rule hiding methods and data hiding methods, they may not avoid negative side impact on nonconfidential portions of the dataset. It follows that two different modified versions of the original dataset may be required to fulfill these two disparate sub-tasks. In this paper, we make attempt to construct only one modified version of the original dataset to fulfill both goals, so that releasing one modified version is sufficient for dual privacy protection. Accordingly, the protection of privacy is simplified with enhanced performance. The target dataset is defined as an unclassified or unlabeled dataset whose feature values are numerical. To achieve this dual privacy protection, a novel technique composed of four schemes is introduced for unsupervised learning. Under the constraints on zero side effect on pattern protection, our implementations can compute some optimal solutions.

The proposed technique is motivated by a unique characteristic of the matrix decomposition techniques, which can provide a compact representation of the data with a reduced-rank while preserving dominant data patterns. Recently, matrix decomposition techniques such as the singular value decomposition (SVD) and nonnegative matrix factorization (NMF) have been proposed for data distortion in PPDM applications [28, 29, 31]. Experimental results in [29] show that NMF can be used to distort sensitive datasets and it outperforms some classical noise-

additive methods in data hiding. It provides a feasible platform to achieve both data hiding and pattern hiding.

NMF can be viewed as a subspace method for basis decomposition [15]. By applying an NMF algorithm to a nonnegative data matrix A of dimension $n \times m$, two nonnegative factor matrices are generated by minimizing the objective functions. Mathematically, this corresponds to factoring the matrix A into two matrices with positive entries, $A \approx HW$. The matrix W has size $k \times m$, with each of the k rows defining a basis vector. The matrix H has size $n \times k$, with each of the n rows defining an additive combination of the basis of the corresponding subject. In [32], an NMF-based clustering algorithm is proposed to cluster text documents. Document corpus is projected into a k -dimensional semantic space and each document is represented as a linear combination of the k topics. The cluster membership of each document can be easily determined by finding the base topic with which the document has the largest projection value. This idea constructs the basic idea for hiding patterns in our approach. It can be assumed that the factor vectors are related to the cluster property of the corresponding subjects. The shift of a subject from one group to other groups may occur whenever the factors are modified.

The remainder of this paper is organized as follows. In Section 2, a description of the problem under study and some definitions are given. The proposed approach is elaborated in Section 3 with four schemes and three related algorithms. Section 4 includes experiments and results. Other related works are reviewed in Section 6. Finally, we conclude the paper in Section 7.

2 Problem Formulation

Our technique targets the simultaneous realization of two subtasks: pattern hiding and data hiding. Pattern hiding is that the data is altered so that it will preserve certain confidential patterns from being discovered, but the influence of data alteration on nonconfidential patterns should be minimized. Data hiding is that the data is modified so that disclosure risk of certain data values is minimized and the influence of data distortion on the mining results is minimized. Only through a single sequence of modifications on the original dataset can these two contradictory goals be achieved simultaneously. In this study, the datasets un-

der consideration are limited to having numerical values and the data patterns under consideration are specified as memberships or relationships of data subjects.

We consider a dataset S consisting of n subjects each of which has m features. Unsupervised learning methods can be used to find the cluster property of the data with a prior assumption of the number of clusters k . S can be partitioned into k subsets called clusters. Each subject is a member of a particular cluster or subset. We can define a binary relation R over the membership of subjects.

Definition 1: Data Model S . Given a dataset S consisting of n independent subjects, with each subject having m numerical features, if we denote the i th subject of S as S_i , then $S = \{S_i\}_{i=1}^n$.

Definition 2: Vector Space Data Model A . Given a data model S , S is represented by a matrix A of dimension $n \times m$ with the rows corresponding to the n subjects and the columns to the m features. If the i th row is denoted by A_i , then A_i represents S_i . The j th feature is represented by the j th column of A , denoted by A^j .

Definition 3: Data Cluster C . Given a dataset of size n with an m -dimensional feature space, $\{S_1, S_2, \dots, S_n\}$, denoted by S , the number of clusters k and a learning algorithm I , C_1, C_2, \dots, C_k are the k subsets, created by I ; let c_1, c_2, \dots, c_k be the k cluster centroids, s.t.

1. $S = C_1 \cup C_2 \cup \dots \cup C_k$,
2. $c_i = \text{mean}(\sum_{S_j \in C_i} S_j)$,
3. $\forall p, q \in \{1, 2, \dots, k\}, C_p \cap C_q = \Phi, p \neq q$, and
4. $\forall i, 1 \leq i \leq n, \exists p, 1 \leq p \leq k, s_i \in C_p$.

Definition 4: Relation R . Given a dataset S , let S^2 denote $S \times S$, the set of all possible ordered pairs of elements of S , a relation R is a binary function $\Psi : (S^2, I, C) \rightarrow \{\text{true}, \text{false}\}$. $\forall (x, y) \in S^2, \exists p, q, 1 \leq p, q \leq k$, s.t. $s_x \in C_p, s_y \in C_q$, and

1. $p = q \rightarrow xRy = \text{true}$,
2. $p \neq q \rightarrow xRy = \text{false}$.

Lemma 1. R is an equivalence relation .

Proof. First, R is reflexive as $\forall s_x \in S, s_i R s_i$. Second, it is symmetric, as $\forall i, j, 1 \leq i, j \leq n, s_i R s_j$ means that s_i

and s_j are in the same cluster which implies $s_j R s_i$. Third, it is transitive, whenever s_i is in the same cluster as s_j is and s_j is in the same cluster as s_t is, then s_i is in the same cluster as s_t is, hence $s_i R s_t$.

Definition 5: Data Modification. Given two datasets A and \tilde{A} , a sequence of modifications is a function $\Psi : (A, F, M) \rightarrow \tilde{A}$ that transforms A into \tilde{A} , where F is the subjects to be modified and M is the modification scheme.

Definition 6: Confidential Relationship Hiding. Let \tilde{S} be the dataset after applying a sequence of modifications on S and a pair $(x, y) \in S^2$. xRy will be hidden if the following conditions are satisfied:

1. $l = xRy$ in S ,
2. $g = xRy$ in \tilde{S} , and
3. $g = \neg(l)$.

3 Proposed Technique

In this section, we describe the proposed dual privacy preserving technique consisting of one data hiding scheme and a set of pattern hiding schemes. All schemes are based on a basic data modification scheme that performs nonnegative matrix factorization (NMF) on the original dataset.

3.1 Basic Data Modification Scheme

Let the original dataset S be encoded by a vector space data model A as in Definition 2. Using some NMF algorithm, A can be decomposed into two nonnegative factor matrices. The scheme can be stated as a transformation from A to \tilde{A} defined as follows: Given a nonnegative data model $A(n \times m)$, find two nonnegative matrices $H(n \times k)$ and $W(k \times m)$ with k being the number of clusters in A , that minimizes $f(A, HW)$, where $f(A, HW)$ is a cost function defining the nearness between the matrices A and HW . The modified version of A is denoted as $\tilde{A} = HW$.

The choice of the cost function f affects the solution of \tilde{A} . Here, the Euclidean distance or the Frobenius norm is

chosen as they are popular in matrix computations,

$$f(A, H, W) = \frac{1}{2} \|A - HW\|_F^2.$$

A standard way to find H and W is by the following least-squares optimization, which minimizes the difference between A and HW :

$$\min_{H, W} f(A, H, W) = \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^m (A(i, j) - (HW)(i, j))^2$$

$$\begin{aligned} \text{subject to } & H(i, a) \geq 0, \\ & W(b, j) \geq 0, \forall i, a, b, j. \end{aligned} \quad (1)$$

NMF algorithms generally begin by initial estimates of the matrices H and W , followed by alternating iterations to improve these estimates. Projected gradient method proposed by Lin [17] will be used in our implementation to directly minimize (1).

After performing basic data modifications on A , the modified dataset is $\tilde{A} = HW$, where

$$H = \begin{bmatrix} H_1 \\ H_2 \\ \vdots \\ H_n \end{bmatrix}, \quad W = \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_k \end{bmatrix}.$$

$$H_i = (h_{i1} \ h_{i2} \ \dots \ h_{is} \ \dots \ h_{ik}), \quad i = 1, 2, \dots, n.$$

$$W_j = (w_{j1} \ w_{j2} \ \dots \ w_{jt} \ \dots \ w_{jm}), \quad j = 1, 2, \dots, k.$$

3.2 Data Hiding Scheme

Based on the basic data modification scheme, data hiding can be easily fulfilled with some simple preprocessing procedure on the original data matrix A . The nonnegative property of A needs to be validated by checking the non-negativity of all entries. Most real-life datasets have non-negative entries. If A has negative entries, its values can be shifted column-wise and then normalized. After this process, \tilde{A} can be generated with the basic data modification scheme. The algorithm is illustrated in Figure 1.

The performance of this scheme is illustrated in Figures 2 and 3. Figure 2 shows the data distributions of a dataset and its modified versions from NMF and two

noise-additive methods. The dataset is synthetically created from three spherical Gaussian distributions and normalized to a nonnegative matrix. It has 100 subjects, each of which has 2 features. Three classes are depicted with three different symbols. The modified version in the upper right subfigure is calculated from an NMF operation with $k = 3$. The lower two subfigures show modified datasets generated from adding normally distributed noise and uniformly distributed noise, respectively. It is clearly observable that the data distributions from NMF and the addition of uniformly distributed noise (lower right) are distorted more than the one from the addition of normally distributed noise (lower left).

Minimizing the impact of data distortion on mining results is another requirement for data hiding schemes. Our basic data modification scheme using NMF can maintain data patterns better than some classical noise-additive methods. The synthetic dataset in Figure 2 is used as an example to demonstrate this claim. In Figure 3, three scatter plots are used to illustrate the execution of a binary Support Vector Machine (SVM) classification on the synthetic data, the modified version using NMF and the modified version using the addition of uniformly distributed noise. A binary SVM classifier is trained to separate class 1 from class 2 and class 3. Using the same training set and testing set, the modified version from NMF has the same correct rate as that of the original data which is 98%. The addition of uniformly distributed noise deteriorates the classification accuracy and its correct rate is reduced to only 54%.

3.3 Pattern Hiding Strategies

Given the number of classes, k , H and W can be calculated. Each row of W represents one of the k clusters. Each of the subjects in S can be represented by an additive combination of the k base vectors.

$$A_i = \sum_{j=1}^k h_{ij} W_j$$

Each element h_{ij} indicates to which degree the subject i belongs to the cluster C_j , while each element w_{ij} represents the degree at which the feature j contributes to the cluster C_i . If the subject i belongs to the cluster C_x , then h_{ix} will take on a larger value than the rest of the elements

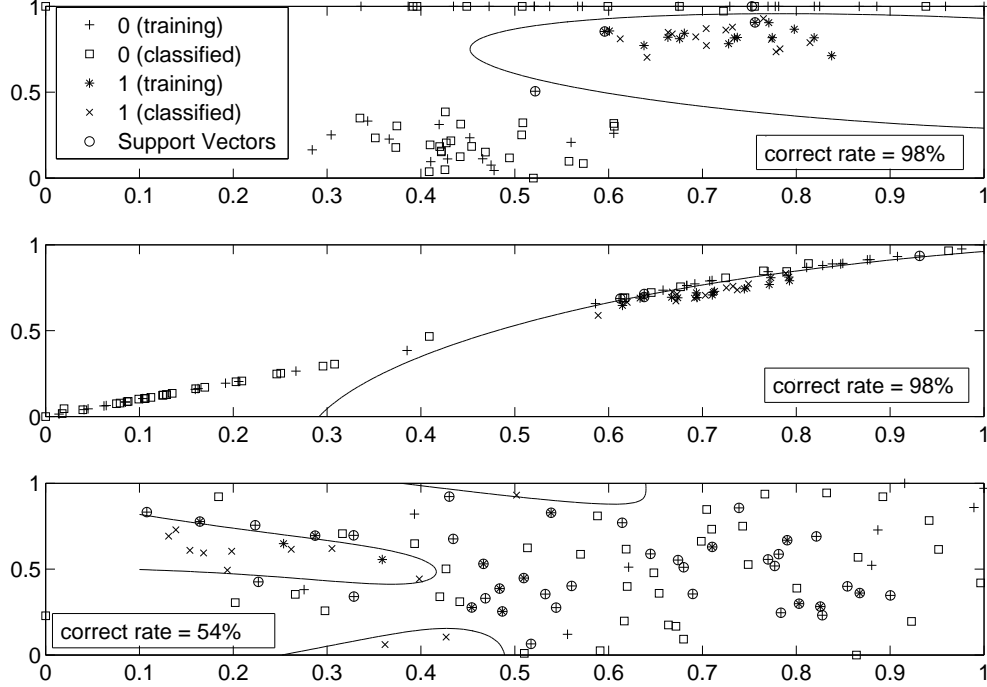


Figure 3: Binary SVM classification on the original data (top), the modified data by NMF (middle) and the modified data by adding uniformly distributed noise (bottom).

in H_i . Therefore, NMF can be viewed as a kind of unsupervised learning that the cluster label of the subjects can be determined by H [32]. The clustering rule is described as: the subject A_x is placed in the cluster C_p if h_{xp} is the largest element in the row of H_x , i.e., $A_x \in C_p$, if $p = \max_j \{h_{xj}\}$. In our experiments, we find that the accuracy of the clustering results by this rule is lower than that of the k-means clustering algorithm. However, this rule implies a underlying fact that any modification on factor vectors may change the memberships of the corresponding subjects. Based on this insight, we design three schemes based on modifying h_{ij} to change the membership of a single subject x or a relationship xRy .

3.3.1 Three Pattern Hiding Schemes

Given a dataset S with k clusters and its vector space model A , H and W are computed using Algorithm 1. The Max-min factor swapping scheme, the factor index swapping scheme and the hybrid modification scheme are described as follows.

Scheme 1: Max-Min Factor Swapping Scheme. Let x be the index of the selected subject in S . The factor vector of S_x is $H_x = (h_{x1} \ h_{x2} \ \dots \ h_{xj} \ \dots \ h_{xk})$. The largest factor is swapped with the smallest factor in H_x . Let $Id_{max} = \max_j \{h_{xj}\}$, $Id_{min} = \min_j \{h_{xj}\}$, $max =$

$h_{xId_{max}}, min = h_{xId_{min}}$, then

$$\begin{aligned} h(x, Id_{max}) &\leftarrow min, \\ h(x, Id_{min}) &\leftarrow max. \end{aligned}$$

Scheme 2: Factor Index Swapping Scheme. Given $(x, y) \in S^2$, i.e., x and y are the indices of one selected subject pair in S . The factor vectors of S_x and S_y are

$$\begin{aligned} H_x &= (h_{x1} \ h_{x2} \dots \ h_{xj} \dots \ h_{xk}), \\ H_y &= (h_{y1} \ h_{y2} \dots \ h_{yj} \dots \ h_{yk}). \end{aligned}$$

Let

$$\begin{aligned} IdX_{max} &= \max_j \{h_{xj}\}, \\ IdY_{max} &= \max_j \{h_{yj}\}, \\ max_y &= h(y, IdY_{max}). \end{aligned}$$

- If S_x and S_y do not have the same index of the maximum factors, i.e., $IdX_{max} \neq IdY_{max}$, we swap the maximum factor of S_y with the factor in the same index as the maximum factor of S_x ,

$$\begin{aligned} h(y, IdY_{max}) &\leftarrow h(y, IdX_{max}) \\ h(y, IdX_{max}) &\leftarrow max_y \end{aligned}$$

- If S_x and S_y have the same index of the maximum factors, i.e., $IdX_{max} = IdY_{max}$, we swap the maximum factor of S_y with any factor other than the maximum factor of S_x . $\exists t, 1 \leq t \leq k, t \neq IdX_{max}$,

$$\begin{aligned} h(y, IdY_{max}) &\leftarrow h(y, t) \\ h(y, t) &\leftarrow max_y \end{aligned}$$

Scheme 3: Hybrid Swapping Scheme. Given $(x, y) \in S^2$, assume that the factor vectors of S_x and S_y are

$$\begin{aligned} H_x &= (h_{x1} \ h_{x2} \dots \ h_{xj} \dots \ h_{xk}), \\ H_y &= (h_{y1} \ h_{y2} \dots \ h_{yj} \dots \ h_{yk}). \end{aligned}$$

Let

$$\begin{aligned} IdX_{max} &= \max_j \{h_{xj}\}, \quad max_x = h(x, IdX_{max}), \\ IdX_{min} &= \min_j \{h_{xj}\}, \quad min_x = h(x, IdX_{min}). \end{aligned}$$

Modify the factor vector of S_y based on S_x by substituting its maximum and minimum factors with those of S_y , then swap them, i.e.,

$$\begin{aligned} h(y, IdX_{max}) &\leftarrow min_x \\ h(y, IdX_{min}) &\leftarrow max_x \end{aligned}$$

3.3.2 Single Membership Hiding

To hide the membership of one subject is identical to a shift of the subject from its source cluster to any other clusters. Since the hiding operation is built on the basic data modification, the uncertainty of NMF computation may lead to different results. Different factor matrices W and H may cause a different shift of the subject even though the same hiding scheme is utilized. In order to improve the predicability on results and take advantage of the flexibility of NMF, we make use of the iteration operations in NMF to find an optimal factorization that fulfills the requirement on data hiding and membership hiding simultaneously. Algorithm 2 in Figure 4 is a description of the single membership hiding scheme. In Algorithm 2, the solution is found with zero side effect on the nonconfidential memberships.

3.3.3 Single-pair Relationship Changing

By Definition 4, the relationship xRy represents whether the subject x and the subject y belong to the same group. In the context of relationship change, there are two possible situations on xRy : from true to false or from false to true. If xRy is negative in the learning result from the modified version of the dataset, then we consider it as a successful hiding. The membership shifts of x and y are not limited. However, any change on other subjects' memberships are not expected, i.e., the side effect should be avoided or limited.

Given a user-specified pair with the confidential relationship, (x, y) in S , the problem can be formulated as $\Psi : (S, (x, y), Scheme) \rightarrow \tilde{A}$. Figure 5 is the proposed procedure to change a single-pair relationship. As to multiple pair relationships, we can rewrite the iteration stopping condition in the algorithm to change the pair relationship one by one.

4 Performance Evaluation

We conduct experiments on the IRIS dataset to evaluate the performance of the proposed technique. IRIS contains 3 classes of 50 subjects each, where each class refers to a type of iris plant and each subject has 4 features. As Figure 6(a) shows, one class in cross marks is linearly separable from the other two in circle and square marks; the latter two classes are not linearly separable from each other. The experiments are mainly designed for evaluating pattern hiding schemes when the k-means clustering is used as a learning tool. For a fair comparison of results, for the k-means clustering in all the experiments, the initial cluster centroids are fixed as the first three data subjects in IRIS.

First, the k-means algorithm is run on IRIS to produce 3 clusters denoted by C_1, C_2, C_3 , 3 centroids denoted by c_1, c_2, c_3 and the corresponding cluster labels. The cluster distribution created from the k-means algorithm is shown in Figure 6(b). C_3 marked in circle contains 50 subjects. C_2 marked in square and C_1 in cross contain 61 and 39 subjects, respectively. 17 subjects are incorrectly grouped and the correct rate is 88.7%. This cluster distribution defined as C_1, C_2, C_3 in Figure 6(b) is considered as the truth for computing clustering accuracy in the subsequent experiments. The following is a description of the truth. To make it clear, the indices are used.

$C_1 = \{101 - 150\} - \{102, 107, 114, 115, 120, 122, 124, 127, 128, 134, 139, 143, 147, 150\} + \{51, 53, 78\}$.

$C_2 = \{51 - 100\} - \{51, 53, 78\} + \{102, 107, 114, 115, 120, 122, 124, 127, 128, 134, 139, 143, 147, 150\}$.

$C_3 = \{1 - 50\}$.

The three cluster centroids are

$$c_1 = [6.8538 \quad 3.0769 \quad 5.7154 \quad 2.0538],$$

$$c_2 = [5.8836 \quad 2.7410 \quad 4.3885 \quad 1.4344],$$

$$c_3 = [5.0060 \quad 3.4180 \quad 1.4640 \quad 0.2440].$$

Then a series of experiments are conducted to evaluate the proposed methods. The experiments abide by a common procedure from the basic data modification to a modified version. The released version is an optimal solution for both data hiding and pattern hiding. As far as learning accuracy and the validation of pattern hiding are concerned, a comparison is made between the truth clustering and the clustering result from a modified dataset.

Measuring the side effect associated with the pattern hiding schemes is a necessary part of the evaluation. An optimal hiding solution should be the one where only user-specified pattern is hidden and all the rest of the patterns are kept intact, i.e., there is no redundant change or nonzero side effect. Because of the assumption that the initial centroids are fixed for all the executions of the k-means algorithm, we can quantify the side effect as a rate of the number of changed subjects among the number of nonconfidential subjects.

For example, in hiding the membership of one subject in IRIS, if 5 other subjects are shifted to clusters different from their original ones, the side effect can be calculated as $5/149$, that is 3.36%. Obviously, the lower the side effect, the better the data usability of a hiding scheme leads to.

The computation of W and H by NMF is implemented by an algorithm in [17]. In our experiments, the tolerance for a relative stopping condition is 10^{-4} . The time limit is 6000 seconds and the iteration limit is 3000.

4.1 Efficiency of Data Hiding Using Basic Data Modification Scheme

A comparison of the basic data modification scheme with two noise-additive data hiding methods is conducted to demonstrate the efficiency of the proposed scheme. One noise-additive method denoted by ND is to add normally distributed noise that is generated with a mean $\mu = 0$ and a standard deviation $\sigma = 0.46$, to the original IRIS dataset. Another method is denoted by UD that adds uniformly distributed noise generated from the interval $[0, 0.8]$ to IRIS. Three modified data versions are produced from the above three data hiding methods. They are denoted by NMF, ND and UD. Then the k-means clustering is run on the three modified datasets. NMF has the highest correct rate of 100%. The correct rates of ND and UD are 86% and 44%, respectively. This result shows that NMF data hiding scheme outperforms ND and UD when data pattern maintenance is concerned. In terms of the accuracy of the data mining algorithm, noise additive methods sometimes degrade learning results [29, 31].

4.2 Membership Hiding Using Scheme 1

In this experiment, Scheme 1 in Section 4.3.1 is evaluated by hiding the membership of the 50th subject. In the truth as defined earlier, the membership of the 50th subject is C_3 . A shift to C_2 or C_1 will hide its original membership. An optimal solution with the minimum side effect can be obtained through the NMF iterations. First, the subject is designed to be shifted to C_2 . One optimal W for this case is computed as:

$$W^* = \begin{bmatrix} 2.4284 & 1.5910 & 0.5626 & 0 \\ 2.0386 & 0.1599 & 2.1913 & 0.5940 \\ 0.6671 & 1.6579 & 0.2504 & 0.5813 \end{bmatrix}.$$

The factor vector of the 50th subject is

$$H_{50} = [1.8918 \quad 0.1394 \quad 0.1679].$$

After swapping its maximum and minimum factor elements by using Scheme 1, the new factor vector is

$$\hat{H}_{50} = [0.1394 \quad 1.8918 \quad 0.1679].$$

Leaving all the other factor vectors unchanged in \hat{H} , an optimal modified version \tilde{A} is constructed as the product of \hat{H} and W^* . When the k-means clustering is run on \tilde{A} , the result is a clean shift of the 50th subject from C_3 to C_2 without any additional membership change in the rest of subjects. That means the side effect is 0%. Therefore, an optimal release dataset can be taken as $\tilde{A}^* = \hat{H}W^*$.

Next, we will make a shift of S_{50} to C_1 . An optimal W generated from the NMF iterations and the corresponding factor vector of S_{50} are as follows:

$$W^{**} = \begin{bmatrix} 1.4285 & 1.1208 & 0.2422 & 0.0210 \\ 1.6549 & 0 & 1.3761 & 0.1504 \\ 1.6739 & 1.2329 & 1.6303 & 0.8675 \end{bmatrix},$$

$$H_{50} = [2.9082 \quad 0.4674 \quad 0.0392].$$

By using Scheme 1, we have

$$\hat{H}_{50} = [0.0392 \quad 0.4674 \quad 2.9082].$$

Accordingly, $\tilde{A}^* = \hat{H}W^{**}$ is an optimal solution for a shift of the 50th subject from C_3 to C_1 . This solution does not bring any other redundant shift so that the rest

of the subjects remain in their original groups. The side effect is 0%.

We also conduct experiments on shifting subjects from C_2 to C_1 or C_3 and from C_1 to C_2 or C_3 . For the 80th subject, one optimal W and the distorted 80th factor vector for the shift from C_2 to C_1 are

$$W^* = \begin{bmatrix} 2.7044 & 0 & 1.7202 & 0 \\ 1.3825 & 0.6344 & 1.4931 & 0.6260 \\ 1.1411 & 0.9137 & 0.1900 & 0.0175 \end{bmatrix},$$

$$\hat{H}_{80} = [1.8403 \quad 1.4754 \quad 0.5700].$$

For the shift of S_{80} from C_2 to C_3 , one optimal solution is

$$W^* = \begin{bmatrix} 0.0284 & 3.2999 & 0 & 0.9529 \\ 1.6979 & 0.9374 & 0.4465 & 0 \\ 1.0185 & 0 & 1.9840 & 0.8098 \end{bmatrix},$$

$$\hat{H}_{80} = [2.6486 \quad 0.0360 \quad 1.1725].$$

For the 130th subject, one optimal solution for the shift from C_1 to C_2 is

$$W^* = \begin{bmatrix} 2.0319 & 0.7374 & 0.8519 & 0 \\ 0.8570 & 1.0289 & 0 & 0.0619 \\ 0.1169 & 0 & 3.4679 & 2.1375 \end{bmatrix},$$

$$\hat{H}_{130} = [0.4487 \quad 3.3424 \quad 0.8188].$$

For the shift of S_{130} from C_1 to C_3 , we have

$$W^* = \begin{bmatrix} 0.1830 & 5.2784 & 0 & 0.8378 \\ 0.9032 & 0 & 3.1744 & 1.4457 \\ 2.6576 & 1.1713 & 0.7117 & 0 \end{bmatrix},$$

$$\hat{H}_{130} = [2.2949 \quad 1.2681 \quad 0.0492].$$

These experimental results show that by using the iteration procedure described in Algorithm 2 in Figure 4, an random optimal solution without any side effect can be computed for membership hidings in IRIS. It demonstrates that Scheme 1 is an effective way to hide confidential memberships. We note that an optimal solution is not unique.

4.3 Relationship Change Using Scheme 2

Given a user-specified pair with the confidential relationship, (x, y) in IRIS, using Scheme 2 to change xRy , the problem is $\Psi : (\text{IRIS}, (x, y), \text{Scheme 2}) \rightarrow \tilde{A}^*$, where \tilde{A}^* is an optimal solution without any side effect.

Test 1: $\Psi : (\text{IRIS}, (50, 80), \text{Scheme 2}) \rightarrow \tilde{A}^*$. $50R80$ is false in the truth clustering of IRIS. We need to find an \tilde{A}^* to change the relationship to true. Scheme 2 is carried out to produce an optimal factorization where the basis matrix is

$$W^* = \begin{bmatrix} 0.1261 & 3.3805 & 0 & 0.8557 \\ 1.7367 & 0.9309 & 0.4587 & 0 \\ 1.4324 & 0 & 2.8763 & 1.1859 \end{bmatrix}.$$

The corresponding factor vectors are

$$H_{50} = [0.1948 \quad 2.8354 \quad 0.0336],$$

$$H_{80} = [0.0496 \quad 2.6134 \quad 0.8012].$$

We may notice that the second elements of both vectors have the largest values, and they should be in the same cluster as the NMF-based clustering rule suggests. The truth here is that they are in the different clusters. Since our aim is to change their relationship, it does not matter how the NMF-based clustering rule suggests, as long as we can negate their existing relationship. Then according to H_{50} and H_{80} , we modify H_{80} by Scheme 2 to get a new factor vector

$$\hat{H}_{80} = [2.6134 \quad 0.0496 \quad 0.8012].$$

Running the k-means clustering on $\tilde{A}^* = \hat{H}W^*$, $50R80$ is changed to TRUE as the membership of the 80th subject is shifted from C_2 to C_3 .

Test 2: $\Psi : (\text{IRIS}, (50, 30), \text{Scheme 2}) \rightarrow \tilde{A}^*$. $50R30$ is true in the truth clustering of IRIS. We need to find an \tilde{A}^* to change the relationship to false. The basis matrix in an optimal factorization is

$$W^* = \begin{bmatrix} 0 & 1.2589 & 0.9849 & 1.2493 \\ 0.5481 & 0 & 0.7449 & 0.2294 \\ 1.1574 & 0.8411 & 0.1990 & 0 \end{bmatrix}.$$

The corresponding factor vectors are

$$H_{50} = [0 \quad 0.8505 \quad 3.9160],$$

$$H_{30} = [0.0650 \quad 1.0059 \quad 3.6315].$$

We then modify H_{30} by Scheme 2 to get a new factor vector

$$\hat{H}_{30} = [3.6315 \quad 1.0059 \quad 0.0650].$$

Running the k-means clustering on $\tilde{A}^* = \hat{H}W^*$, $50R30$ is changed to false as the membership of the 30th subject is shifted from C_3 to C_2 .

Test 3: $\Psi : (\text{IRIS}, (50, 30), (80, 130), \text{Scheme 2}) \rightarrow \tilde{A}^*$. In this experiment, two confidential relationships are specified as $50R30$ and $80R130$. $50R30$ is true and $80R130$ is false in the truth clustering of IRIS. An \tilde{A}^* is required to negate these two relationships. Compared to the previous two experiments, the number of iterations increases. After 16 iterations, an optimal factorization is found as

$$W^* = \begin{bmatrix} 0.2297 & 1.1217 & 1.7552 & 1.5272 \\ 1.3011 & 0.9201 & 0.2528 & 0 \\ 2.5082 & 0.7222 & 2.0846 & 0.5569 \end{bmatrix}.$$

H_{30} and H_{130} are modified based on Scheme 2. The modified factor vectors are

$$\hat{H}_{30} = [3.0946 \quad 0.0978 \quad 0.2770],$$

$$\hat{H}_{130} = [0.2837 \quad 2.3770 \quad 0.9609].$$

Then the k-means clustering is run on $\tilde{A}^* = \hat{H}W^*$, $50R30$ is changed to false as the membership of the 30th subject is shifted from C_3 to C_2 . $80R130$ is changed to true as the membership of the 130th subject is shifted from C_1 to C_2 . The solution is not unique, however, the following solution is generated after 77 iterations:

$$W^* = \begin{bmatrix} 1.2481 & 1.5489 & 1.6029 & 1.1703 \\ 2.1535 & 0.2067 & 2.1337 & 0.5170 \\ 1.6640 & 1.1971 & 0.3128 & 0 \end{bmatrix}.$$

The above three experiments indicate the viability of Scheme 2 in changing subject relationships. Similar to the membership hiding, in our experiments, an optimal solution has always been obtainable with zero side effect.

4.4 Relationship Change Using Scheme 3

In this section, the experiments are to examine the effectiveness of Scheme 3 on solving the problem defined as $\Psi : (\text{IRIS}, (x, y), \text{Scheme 3}) \rightarrow \tilde{A}^*$, where \tilde{A}^* is an optimal solution without any side effect. In order to make a comparison with Scheme 2, the three experiments are executed under the same conditions as in Section 4.3.

Test 1: $\Psi : (\text{IRIS}, (50, 80), \text{Scheme 3}) \rightarrow \tilde{A}^*$. Scheme 3 is carried out to distort the factor vector of H_{80} . An optimal solution is generated after 6 iterations, where the 80th subject is moved from C_2 to C_3 and $50R80$ becomes true in the clustering result on the distorted dataset.

$$W^* = \begin{bmatrix} 2.9125 & 2.3836 & 0.3245 & 0 \\ 0.9380 & 0.1511 & 0.7462 & 0.1220 \\ 0 & 3.5909 & 1.5772 & 2.7915 \end{bmatrix}.$$

The two corresponding factor vectors are

$$H_{50} = [1.2916 \quad 1.3134 \quad 0.0083],$$

$$H_{80} = [0.6076 \quad 4.1582 \quad 0.1534].$$

The distorted H_{80} by Scheme 3 is

$$\hat{H}_{80} = [0.6076 \quad 0.0083 \quad 1.3134].$$

Test 2: $\Psi : (\text{IRIS}, (50, 30), \text{Scheme 3}) \rightarrow \tilde{A}^*$. One \tilde{A}^* is found. By running the k-means clustering on $\tilde{A}^* = \hat{H}W^*$, the membership of the 30th subject is shifted from C_3 to C_1 . $50R30$ is changed to FALSE. The basis matrix in the solution is

$$W^* = \begin{bmatrix} 1.3317 & 0.6553 & 0.3877 & 0 \\ 0.5512 & 1.4534 & 0 & 0.2278 \\ 1.0108 & 0.0979 & 1.9947 & 0.8511 \end{bmatrix}.$$

The two factor vectors are

$$H_{50} = [3.4230 \quad 0.7278 \quad 0.0373],$$

$$H_{30} = [3.1115 \quad 0.7687 \quad 0.1648].$$

We distort H_{30} by Scheme 3 as

$$\hat{H}_{30} = [0.0373 \quad 0.7687 \quad 3.4230].$$

Test 3: $\Psi : (\text{IRIS}, (50, 30), (80, 130), \text{Scheme 3}) \rightarrow \tilde{A}^*$. After just 2 iterations, an optimal factorization is produced as

$$W^* = \begin{bmatrix} 1.0557 & 0 & 2.0637 & 0.8439 \\ 0.0048 & 2.5042 & 0 & 0.7697 \\ 1.5353 & 0.8594 & 0.4032 & 0 \end{bmatrix}.$$

The related factor vectors are

$$H_{30} = [0.1599 \quad 0.2412 \quad 2.9743],$$

$$H_{50} = [0.0480 \quad 0.2108 \quad 3.2206],$$

$$H_{80} = [1.1291 \quad 0.0318 \quad 2.9315],$$

$$H_{130} = [2.1085 \quad 0.0393 \quad 3.2880].$$

H_{30} and H_{130} are modified based on Scheme 3. The modified factor vectors are

$$\hat{H}_{30} = [3.2206 \quad 0.2412 \quad 0.0480],$$

$$\hat{H}_{130} = [2.1085 \quad 2.9315 \quad 0.0318].$$

The k-means clustering is run on $\tilde{A}^* = \hat{H}W^*$, $50R30$ is changed to false as the membership of the 30th subject is shifted from C_3 to C_2 . $80R130$ is changed to true as the membership of the 130th subject is shifted from C_1 to C_2 .

Through these three experiments, we show that Scheme 3 can change specified relationships as Scheme 2 does. By setting a stopping condition with which the side effect is zero, an optimal solution can be computed and it is not unique. We note that multiple relationship hiding does not necessarily take more time than the single relationship hiding.

5 Related Work

Statistical disclosure control (SDC) is one of the earliest field in data privacy preservation. The problem of protecting sensitive information in a database while allowing statistical queries has been studied extensively since the late 1970's [1, 24]. Early in 1989, Adam and Wortmann [1] conducted a comprehensive survey on security-control methods for SDC. The methods are classified under four general approaches: conceptual, query restriction, data perturbation, and output perturbation. The survey introduced probability-distribution perturbation and fixed-data

perturbation approaches. For the fixed-data perturbation approach, the year of 1984 saw Traub *et al.* [25] developed an additive-perturbation method for numerical attributes by adding or multiplying a random variable to a true value. It might be the first randomization scheme in privacy protection. With the same idea as probability-distribution method [1], several reconstruction-based or randomization-based methods by adding some noise to the original data have been widely used for privacy protection [9, 20]. The simplest version is noise-additive approach [2, 10, 14]. The approach is intuitive and easy to understand, however, researchers have recently identified privacy breaches as one of its major problems [14, 34]. A filtering method is proposed based on random matrix theory to reconstruct private data from the randomized dataset [13]. It shows that randomization preserves little privacy in many cases. Two other data reconstruction methods, Principal component analysis-based and Bayes estimate-based, are proposed in [10] to restore original data from disturbed data. It is suggested that the amount of private information that can be disclosed is related to data correlation, and the more the correlation of the noises resembles that of the original data, the better privacy preservation can be achieved [10].

The later comer in data perturbation category is random projection approach, most of which are multiplicative perturbation in the context of computing inner product matrix [18, 19]. These methods are based on the Johnson Lindenstrauss lemma [11], which places bounds on Euclidean distance distortion due to any dimensionality reduction transform.

The more recent approach is based on the data matrix-decomposition strategies [33]. The use of singular value decomposition (SVD) technique for data distortion is proposed in [31]. In [28], SVD is used to distort selected portions of the datasets, and sparsification techniques by removing small size entries in the approximates furthers data distortion level. NMF is proposed to distort sensitive datasets and enables accurate classification on distorted datasets [29].

Besides these methods based on distorting the original data values, Clifton *et al.* suggest another class of approaches to modify data mining algorithms so that they allow data mining operations on distributed datasets without knowing the exact values of the data or without direct accessing the original data [6].

For association rule hiding, two approaches based on heuristic modification are adopted to prevent association rules from being generated [8]. One is to hide the frequent sets from which rules are derived. The second is to reduce their importance by setting their confidence below a user-specified threshold. Verykios *et al.* [27] present five algorithms to hide sensitive association rules by insertion or removal of records. Three of them belong to the first approach that decreases either the confidence or the support of a set of sensitive rules until the rules are hidden. The other two use the second approach to decrease the support of a set of large itemsets until it is below a user-specified threshold so that no rule can be derived from the selected itemsets. However, the approaches make a strong assumption of no overlapping, i.e., all the items in a sensitive rule do not appear in any other sensitive rule. Some undesirable side effects cannot be avoided, such as lost rules (nonsensitive rules falsely hidden) and ghost rules (spurious rules falsely generated). In order to limit side effects, Wu *et al.* [30] propose heuristic methods for increasing the number of hidden sensitive rules and reducing the number of modified entries. Atallah *et al.* [5] use an itemset graph to hide sensitive itemsets referred to as data sanitization.

For classification rule hiding, a reconstruction-based framework for categorical datasets is proposed by Natwichai *et al.* [21, 22]. After extracting sensitive rules, a new decision tree is built on nonsensitive subset of rules. A new dataset is generated from the decision tree. The authors claim that even though the difference in representation between the new and original datasets can be found, the approach can maintain high level data usability.

Nonnegative matrix factorization (NMF) is popular for approximating nonnegative data in a parts-based context. NMF is from positive matrix factorization (PMF) developed by Juvela *et al.*, and later becomes popular in the computational science community [12]. Interest in NMF increased when a fast algorithm, based on iterative update, was developed by Lee and Seung [16], particularly as they were able to show that it produced intuitively reasonable factorizations for a face recognition problem. NMF facilitates the analysis and classification of data. They also found NMF to be a useful tool in text data mining [15]. NMF has recently been shown to be a very useful technique in approximating high dimensional data with nonnegative components. Xu *et al.* [32] demonstrated that

NMF-based indexing outperforms traditional vector space approaches to information retrieval such as latent semantic indexing for document clustering on a few benchmark test collections.

6 Conclusion

In this paper, we present a novel technique to achieve simultaneous realization of data hiding and pattern hiding. One scheme is proposed to achieve basic data distortion by way of nonnegative matrix factorization. Three schemes are designed to slightly modify the related factors based on a modified dataset generated from NMF. The attractive advantage of the proposed technique is that a single modified version satisfies both of the two contradictory goals. On one hand, matrix factorization provides a good approximation of the original datasets. That supports our technique for distortion on the data values and comparable mining accuracy. On the other hand, taking advantage of an underlying correlation of the factor vectors with cluster properties in the unsupervised learning, our technique is capable of hiding confidential patterns while keeping intact nonconfidential patterns. Practically, the merit of our technique is derived from the fact that one released data version can provide dual protection on general data and specified patterns. The strength and efficiency of privacy protection is enhanced. Empirical evaluation on the IRIS dataset indicates that our technique is an attractive solution to a combined hiding of data values and patterns. In particular, an optimal solution without any undesirable side effect can be easily computed as far as some particular constraints are imposed on the NMF iterations. Our preliminary results show the promising significance of NMF on privacy preserving data mining. More experiments are needed to test the robustness and scalability of this technique on other datasets of larger size. In addition, extension of our approach to supervised learning is warranted.

References

- [1] N. R. Adam and J. C. Worthmann. Security-control methods for statistical databases: a comparative study. *ACM Computing Surveys*, 21(4):515–556, 1989.
- [2] C. C. Aggarwal and P. S. Yu. A condensation approach to privacy preserving data mining. In *Advances in Database Technology - EDBT 2004, the 9th International Conference on Extending Database Technology*, pages 183–199, Heraklion, Crete, Greece, March 2004.
- [3] D. Agrawal and C. C. Aggarwal. On the design and quantification of privacy preserving data mining algorithms. In *Symposium on Principles of Database Systems*, 2001.
- [4] R. Agrawal and R. Srikant. Privacy-preserving data mining. In *Proceedings of the ACM SIGMOD Conference on Management of Data*, pages 439–450, Dallas, Texas, May 2000. ACM Press.
- [5] M. Atallah, E. Bertino, A. Elmagarmid, M. Ibrahim, and V. Verykios. Disclosure limitation of sensitive rules. In *Proceedings of the 1999 Workshop on Knowledge and Data Engineering Exchange*, pages 45–52, 1999.
- [6] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, and M. Zhu. Tools for privacy preserving distributed data mining. *ACM SIGKDD Explorations*, 4(2):1–7, 2003.
- [7] C. Clifton and D. Marks. Security and privacy implication of data mining. In *Proceedings of the Workshop on Data Mining and Knowledge Discovery*, number 96-08, pages 15–19, Montreal, Canada, June 1996. University of British Columbia Department of Computer Science.
- [8] E. Dasseni, V. S. Verykios, A. K. Elmagarmid, and E. Bertino. Hiding association rules by using confidence and support. In *Lecture Notes In Computer Science; Vol. 2137, Proceedings of the 4th International Workshop on Information Hiding*, pages 369–383. Springer-Verlag, April 2001.
- [9] A. V. Evfimievski, J. Gehrke, and R. Srikant. Limiting privacy breaches in privacy preserving data mining. In *PODS*, pages 211–222, 2003.
- [10] Z. Huang, W. Du, and B. Chen. Deriving private information from randomized data. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pages 37–48, Baltimore, MD, June 2005.
- [11] W. Johnson and J. Lindenstrauss. Extensions of Lipschitz mapping into Hilbert space. *Contemporary Mathematics*, 26:189–206, 1984.
- [12] M. Juvela, K. Lehtinen, and P. Paatero. The use of positive matrix factorization in the analysis of molecular line spectra from the thumbprint nebula. In *Proceedings of the Fourth Haystack Conference on Clouds, Cores and Low Mass Stars*, volume 65, pages 176–180. Astronomical Society of the Pacific Conference Series, 1994.
- [13] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar. On the privacy preserving properties of random data perturbation techniques. In *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM 2003)*, pages 99–106. IEEE Computer Society, 2003.

- [14] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar. Random-data perturbation techniques and privacy-preserving data mining. *Knowledge and Information System*, 7(4):387–414, 2005.
- [15] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401:788–791, 1999.
- [16] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. *Advances in Neural Information Processing Systems*, 13:556–562, 2001.
- [17] C. Lin. Projected gradient methods for non-negative matrix factorization. *Neural Computation*, to appear, 2007.
- [18] K. Liu, H. Kargupta, and J. Ryan. Random projection-based multiplicative data perturbation for privacy preserving distributed data mining. *IEEE Transactions on Knowledge and Data Engineering*, 18(1):92–106, 2006.
- [19] K. Muralidhar, D. Batra, and P. J. Kirs. Accessibility, security and accuracy in statistical databases: the case for multiplicative fixed data perturbation approach. *Management Science*, 9(4):1549–1564, 1995.
- [20] K. Muralidhar and R. Sarathy. Security of random data perturbation methods. *ACM Transactions on Database Systems*, 24(4):487–493, 1999.
- [21] J. Natwichai, X. Li, and M. E. Orłowska. Hiding classification rules for data sharing with privacy preservation. In *Proceedings of the 7th International Conference on Data Warehousing and Knowledge Discovery (DaWak 2005)*, pages 468–477, August 2005.
- [22] J. Natwichai, X. Li, and M. E. Orłowska. A reconstruction-based algorithm for classification rules hiding. In *Database Technologies 2006, Proceedings of the 17th Australasian Database Conference*, pages 49–58, Hobart, Tasmania, Australia, January 2006.
- [23] D. O’Leary. Knowledge discovery as a threat to database security. In *Proceedings of the First International Conference on Knowledge Discovery and Databases*, pages 507–517, 1991.
- [24] A. Shoshani. Statistical databases: Characteristics, problems, and some solutions. In *Proceedings of Eighth International Conference on Very Large Data Bases*, pages 208–222, Mexico City, Mexico, September 1982. Morgan Kaufmann.
- [25] J. F. Traub, Y. Yemini, and H. Wozniakowski. The statistical security of a statistical database. *ACM Transactions on Database Systems*, 9(4):672–679, 1984.
- [26] V. S. Verykios, E. Bertino, I. Fovino, L. Provenza, Y. Saygin, and Y. Theodoridis. State-of-the-art in privacy preserving data mining. *SIGMOD Record*, 33(1):50–57, 2004.
- [27] V. S. Verykios, A. K. Elmagarmid, E. Bertino, Y. Saygin, and E. Dasseni. Association rule hiding. *IEEE Transaction on Knowledge and Data Engineering*, 16(4):414–447, 2004.
- [28] J. Wang, J. Zhang, S. Xu, and W. Zhong. A novel data distortion approach via selective SSVD for privacy protection. *International Journal of Information and Computer Security*, 2007. to appear.
- [29] J. Wang, W. Zhong, and J. Zhang. NNMF-based factorization techniques for high-accuracy privacy protection on non-negative-valued datasets. In *Proceedings of the 2006 IEEE Conference of Data Mining, International Workshop on Privacy Aspects of Data Mining*, pages 513–517. IEEE Computer Society, 2006.
- [30] Y. H. Wu, C. M. Chiang, and A. L. Chen. Hiding sensitive association rules with limited side effects. *IEEE Transaction on Knowledge and Data Engineering*, 19(1):29–42, 2007.
- [31] S. Xu, J. Zhang, D. Han, and J. Wang. Singular value decomposition based data distortion strategy for privacy protection. *Knowledge and Information Systems*, 10(3):383–397, 2006.
- [32] W. Xu, X. Liu, and Y. Gong. Document clustering based on non-negative matrix factorization. In *Proceedings of SIGIR’03*, pages 267–273, Toronto, Canada, July 2003.
- [33] J. Zhang, J. Wang, and S. Xu. *Matrix decomposition-based data distortion techniques for privacy preservation in data mining*. Technical Report TR 472-07, Department of Computer Science, University of Kentucky, KY, USA, 2007.
- [34] N. Zhang, S. Wang, and W. Zhao. A new scheme on privacy-preserving data classification. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 374–383, Chicago, IL, August 2005.

Algorithm 1. Data Hiding Scheme**INPUT:** a dataset A , the number of classes k **OUTPUT:** a modified version \tilde{A} , two factor matrices: H and W

1. NonNeg = 1
2. For each entry of A , $A(i, j)$, do
3. If $A(i, j) < 0$, then NonNeg = 0;
4. If NonNeg == 0
5. do nonnegativity normalization on A
6. Compute H and W
7. Calculate $\tilde{A} = HW$

Figure 1: Data hiding scheme.

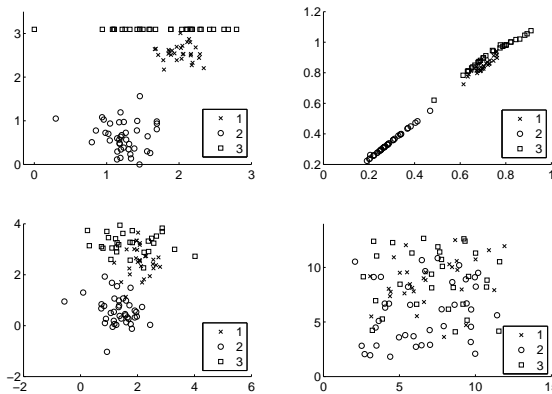


Figure 2: Distributions of a 2D synthetic dataset with 3 classes and its modified version from NMF are in the upper two subfigures. The bottom two subfigures are distributions of modified data using the two noise-additive methods.

Algorithm 2. Single Membership Hiding**INPUT:** a dataset S with its vector space model A , cluster truth C , the index x of the confidential subject, the old membership of S_x , the new membership of S_x .**OUTPUT:** a modified version \tilde{A} , two factor matrices: H and W , one distorted version of H (denoted by \hat{H})**BEGIN**

1. Set $Label$ = the old membership of S_x
2. DO iteration WHILE ($Label \neq$ the new membership of S_x) || $sideEffect \neq 0$)
3. use Algorithm 1 to generate H and W .
4. modify the factor vector H_x of S_x by Scheme 1 to produce \hat{H}
5. compute a modified version of $\tilde{A} = \hat{H} * W$
6. do learning process on \tilde{A} to get new class labels
7. $Label \leftarrow$ the new class label of S_x
8. check other subjects' membership shifts.
9. update $sideEffect$
10. END

END

Figure 4: Single membership hiding scheme.

Algorithm 3. Single-pair Relationship Changing

INPUT: a dataset S with its vector space model A , cluster truth C , a pair (x,y) with a confidential relationship: xRy

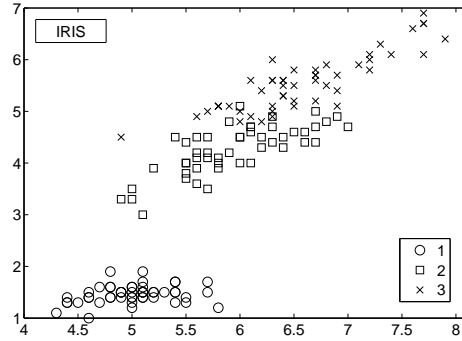
OUTPUT: a modified version \tilde{A} , two factor matrices: H and W , one distorted version of H (denoted by \hat{H})

BEGIN

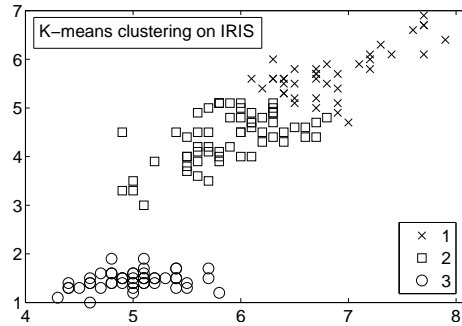
1. Set $pairTruth = xRy$
2. Set $pairNOT = \neg xRy$
3. DO iteration WHILE ($pairNOT == pairTruth$ || $sideEffect != 0$)
4. use Algorithm 1 to generate H and W .
5. modify the factor vectors: H_x or H_y by Scheme 2 or Scheme 3 to produce \hat{H}
6. compute a modified version of $\tilde{A} = \hat{H} * W$
7. do learning process on \tilde{A} to get new class labels
8. $pairNOT \leftarrow (xRy)_{new}$
9. check other subjects' membership shifts.
10. update $sideEffect$
11. END

END

Figure 5: Single-pair relationship change scheme.



(a) Cluster distribution of IRIS.



(b) K-means clustering on IRIS.

Figure 6: IRIS dataset and cluster distribution.