Abstract — Powerful modern access to huge amounts of various data having high or low level of privacy brings out a concurrent increasing demand for preserving data privacy. The challenge is how to protect attribute values without jeopardizing the similarity between data objects under analysis. In this paper, we further our previous work on applying matrix decomposition techniques to protect privacy and to present a novel algebraic technique based on iterative methods for non-negative-valued data distortion. As an unsupervised learning method for uncovering latent features in high-dimensional data, a low rank nonnegative matrix factorization (NMF) is used to preserve natural data non-negativity and avoid subtractive basis vector and encoding interactions present in techniques such as principal component analysis. This is the first in privacy preserving data mining to combine non-negative matrix decomposition with data distortion processing. Two iterative methods to solve bound-constrained optimization problem in NMF are compared with experiments on Wisconsin Breast Cancer Dataset. The overall performance of NMF on distortion level and data utility is compared to our previously proposed SVD-based distortion strategies and other existing popular data perturbation methods. Data utility is examined by cross validation of a binary classification using support vector machine. Our experimental results indicate that, in comparison with standard data distortion techniques, the proposed NMF-based method is very efficient in balancing data distortion and data utility, and it affords a feasible solution with a good promise on high-accuracy privacy preserving data mining.

Index Terms — non-negative matrix factorization, privacy, iterative method, data mining

1. INTRODUCTION

The availability of large scale computing platforms and instrumentation for data collection have created extremely large data repositories that can be utilized through data mining tools. A trade-off between sharing confidential information for analysis and keeping individual, corporate and countries privacy is a growing challenge. It has motivated a great deal of research aimed to answer the following questions: how can data be exchanged securely for cooperative analysis or outsourcing analysis? How can important structure and underlying patterns be found within a large data set? How and when can this hidden structure be used to help predict missing data or to clean data that is imprecise or partially incorrect [1]? The increasing concern on privacy and related research brings out a new branch, known as privacy preserving data mining (PPDM).

The general goal of our work is defined as to hide to the outside world sensitive individual data, and simultaneously preserve the underlying data pattern and semantics so that the construction of a decision model on distorted data is enabled and it is equivalent to or even better than the model using the original data from the viewpoint of decision accuracy [2]. A desirable solution must consider not only privacy safeguards, but also accurate mining results.

The input in a data mining task can be represented by a vector space model, where a collection of records or objects is encoded as an $n \times m$ object-attribute matrix. Datasets of interest often lead to a very high dimensional matrix representation [3]. Moreover, it is observable that many real-world datasets have non-negative values for attributes. In fact, any existing data distortion methods inevitably fall into the
context of matrix computation. For instance, having the longest history in privacy protection area and by adding random noise to the data, additive noise methods can be viewed as a random matrix and therefore its properties can be understood by studying the properties of random matrices [4, 5].

Matrix decomposition in numerical linear algebra typically serves the purpose of finding a computationally convenient means for obtaining a solution to the original linear system. In contrast to its usage as a mechanism for obtaining another end, within the field of data mining, matrix decomposition also plays a major role but usually not just for the purpose of solving systems of equations. In this context its major purpose is to obtain some form of simplified low-rank approximation to original dataset for understanding the structure of data, particularly the relationship within the objects and within the attributes and how the objects relate to the attributes [10]. Our previous work studied using matrix decomposition techniques on privacy-preserving data mining [11]. A unique characteristic of matrix decomposition, a compact representation with reduced-rank while preserving dominant data patterns, stimulates our attempt on utilizing it to realize a two-win task both on high level privacy and high accuracy data utility.

In our previous work [2, 11], a set of hybrid methods that combines Singular Value decomposition (SVD) and sparsification strategies [12] was proposed. It has been experimentally shown that the application of matrix decomposition techniques is one of the feasible channels to better results on privacy protection and higher accuracy than additive noise methods for high accuracy privacy preservation classification. With our previous work on matrix decomposition in [6, 7, 8, 9], we recently investigated on using matrix decomposition techniques to achieve the general goal claimed above: high-accuracy privacy preservation.

Our current study is focused on the context of classifying objects from large non-negative-valued datasets. For this framework, taking advantage of matrix theory and powerful computing capability of iterative methods, the main objective is to provide an efficient and flexible technique for an error-bounded approximation of non-negative-valued datasets. Our proposed method has two important aspects: (i) non-negative matrix factorization (NMF) is adapted to provide a least-squares compression version of original datasets. (ii) By using iterative methods, solving least-squares optimization problem provides an attractive flexibility for data administrators to tailor our solution according to their specific requirement.

The remaining part of the paper is organized as follows. Section II offers an overview of the related work and the application of matrix decomposition techniques in privacy preserving data mining. Iterative implementation of non-negative matrix factorization and description of the proposed algorithm are presented in Section III. Measurement on data privacy and data utility is defined in Section IV. Section V discusses performance of our method in terms of data privacy and data utility and makes a comparison with other privacy protection methods. Finally, we conclude our work with a summary of results and outlines of ongoing research in Section VI.

II. BACKGROUND AND RELATED WORK

Intuitively there are three ideas on disguising sensitive data. One is to transform original data into protected, publishable data by data perturbation. An alternative is to generate a new dataset (synthetic dataset), not from the original data, but from random values that are adjusted in order to have the same feature pattern as the original data. A third possibility is to build a hybrid dataset as a mixture of a distorted one and a synthetic one [13]. Most methods in literature are based on element-wise random perturbation.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
<th>Symbol</th>
<th>Meaning</th>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$A$</td>
<td>Original matrix</td>
<td>$A_{(i,j)}$</td>
<td>$i^{th}$ entry of $A$</td>
<td>$A_{(i)}$</td>
<td>$i^{th}$ row of matrix $A$</td>
</tr>
<tr>
<td>$n$</td>
<td>Number of rows</td>
<td>$m$</td>
<td>Number of columns</td>
<td>$A_{i}$</td>
<td>Submatrix of $A$</td>
</tr>
<tr>
<td>$\tilde{A}$</td>
<td>Approximation matrix of $A$</td>
<td>$A^{(k)}$</td>
<td>Rank-$k$ approximation of matrix</td>
<td>$A^T$</td>
<td>Transpose of $A$</td>
</tr>
<tr>
<td>$|\cdot|_F$</td>
<td>Frobenius-norm of a matrix</td>
<td>$|\cdot|_2$</td>
<td>2-norm of a vector</td>
<td>$\sigma$</td>
<td>Eigenvalue / singular value</td>
</tr>
</tbody>
</table>

A. Matrix Approximation

In practice, problems often involve fifty or a hundred attributes. We might typically believe that each feature is useful for at least some of the discriminations; while we may doubt that each feature provides
Iterative Matrix Factorization Techniques for High-Accuracy Privacy Protection on Non-negative-valued Datasets

independent information, intentionally superfluous features have not been included. The most important is how classification accuracy depends upon the dimensionality and amount of training data and the second is the computational complexity of designing the classifier [14].

Therefore, finding an error-bounded low-rank approximation of real dataset is always an essential subtask. It can be very useful in many problems where distance computations and comparisons are needed, because in high dimensions distance computations are very slow and moreover it is known that the distance between almost all pairs of points is the same with high probability and almost all pairs of points are orthogonal [46].

The Johnson-Lindenstrauss lemma [16] shows that a set of \( n \) points in high dimensional Euclidean space can be mapped down into an \( O(\log n/\varepsilon^2) \) dimensional Euclidean space such that the distance between any two points changes by only a factor of \( (1 \pm \varepsilon) \). In other words, a data set of \( n \) points can be embedded in a subspace of dimension \( O(\log n) \) with little distortion on the pair-wise distances. Random projection and Singular Value Decomposition (SVD) are two popular dimensionality reduction techniques. We instead consider non-negative matrix factorization in this paper.

B. Non-negative matrix factorization (NNMF)

The idea of positive matrix factorization is developed by Paatero at the University of Helsinki, and becomes popular in the computational science community [17]. Interest in positive matrix factorization increased when a fast algorithm for Non-negative Matrix Factorization (NNMF), based on iterative update, was developed by Lee and Seung [18], particularly as they were able to show that it produced intuitively reasonable factorizations for a face recognition problem, and NNMF facilitates the analysis and classification of data from image or sensor articulation databases made up of images showing a composite object in many articulations, poses, or observation views. They also found NMF to be a useful tool in text data mining [19]. In the past few years, several papers have discussed NNMF techniques and successful applications to various databases where the data values are non-negative [20]. NNMF has recently been shown to be a very useful technique in approximating high dimensional data where the data are comprised of non-negative components [21, 22, 23, 24, 25, 26].

III. ITERATIVE DATA DISTORTION STRATEGY

Truncated singular value decomposition can be viewed as a weighted summation of rank-one approximations to a sequence of matrices. The associated weights are the corresponding singular values.

\[
A^{(i)} = \sum_{i \in \mathbb{N}} \sigma_i U(i) V(i)^T
\]

Our previous work shows that SVD is a good solution for data protection of high accuracy classification. However, a drawback is associated with extraction of singular vector of orthogonal decompositions. If the underlying data only consists of nonoverlapping, i.e. orthogonal patterns, SVD performs successfully. If patterns with similar strengths overlap, attributes contained in some of the previously discovered patterns are extracted from each pattern. In orthogonalizing the second vector with respect to the first vector, SVD introduces negative values into the second vector. There is no easy interpretation of these negative values in the context of most data mining tasks, and negative components contradict physical realities.

Considering the non-negative-valued characteristic of most datasets, a nonorthogonal decomposition that does not introduce negative values into the component vectors is required. In the paper, non-negative matrix factorization is used to distort original dataset.

NNMF is a vector space method to obtain a representation of data using non-negative constraints. These constraints can lead to a parts-based representation because they allow only additive, not subtractive, combinations of the original data. This is in contrast to techniques for finding a reduced dimensional representation based on SVD [20].
A. Non-negative Matrix Factorization

Given a nonnegative matrix $A \in \mathbb{R}^{n \times m}$ with $A(i,j) \geq 0$ and a pre-specified positive integer $k < \min\{n, m\}$, NNMF finds two nonnegative matrices $W \in \mathbb{R}^{n \times k}$ with $W(i,j) \geq 0$ and $H \in \mathbb{R}^{k \times m}$ with $H(i,j) \geq 0$ so that $A = WH$ that minimizes the objective function

$$f(W,H) = \frac{1}{2} \|A - WH\|^2_F$$  \hspace{1cm} (2)

The usual way to find $W$ and $H$ is by the following least-squares optimization, which minimizes the difference between $A$ and $WH$:

$$\min_{W,H} f(W,H) = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{m} (A(i,j) - (WH)(i,j))^2$$  \hspace{1cm} (3)

subject to $W_{ia} \geq 0, H_{bj} \geq 0, \forall i,a,b,j$.

In optimization, inequality’s upper- and lower-bounding variables are referred to as bound constraints. Hence (3) is a standard bound-constrained optimization problem. There are several methods to solve (3) in literature. Algorithms designed to approximate $A$ generally begin by initial estimates of the matrices $W$ and $H$, followed by alternating iterations to improve these estimates.

B. First Order Optimality Condition

The gradient of $f(W,H)$ at $(W,H)$ can be expressed as two partial derivatives to elements in $W$ and $H$ respectively.

$$\nabla f(W,H) = (\nabla_W f(W,H), \nabla_H f(W,H))$$

\begin{align*}
\nabla_W f(W,H) &= (WH - A)H^T \\
\nabla_H f(W,H) &= W^T (WH - A)
\end{align*}

By a corollary in [27], at a solution $(W,H)$ of the non-negative matrix factorization problem, it is necessary that

$$(A - WH)H^T \leq 0 \quad W^T (A - WH) \leq 0$$  \hspace{1cm} (5)

Therefore, $(W,H)$ is a stationary point if and only if

$$W_{ia} \geq 0, H_{bj} \geq 0, \nabla_W f(W,H)_{ia} \geq 0, \nabla_H f(W,H)_{bj} \geq 0, \quad W_{ia} \nabla_W f(W,H)_{ia} = 0, H_{bj} \nabla_H f(W,H)_{bj} = 0, \forall i,a,b,j.$$  \hspace{1cm} (6)

C. Multiplicative update algorithm

Multiplicative update algorithm is the most popular approach proposed by Lee and Seung [18]. The iterative algorithm updates each entry of $W$ and $H$ on each iterative step.

\begin{algorithm}[H]
1. Initialize $W$ and $H$ and scale the columns of $W$ to unit norm. $W_{ia} > 0, H_{bj} > 0, \forall i,a,b,j$.
2. In each step, $k=1,2,\ldots$

\begin{equation*}
W_{ia}^{k+1} = W_{ia}^k \frac{(A(H^k)^T)^T_{ia}}{(W^k H^k (H^k)^T)^T_{ia}}, \forall i,a.
\end{equation*}

\end{algorithm}
Iterative Matrix Factorization Techniques for High-Accuracy Privacy Protection on Non-negative-valued Datasets

\[ H_{bj}^{k+1} = H_{bj}^k \left( \frac{(W_k^{k+1})^T A_{bj}}{(W_k^{k+1})^T W_k^{k+1} H^k_{bj}} \right), \forall b, j. \]  

(8)

It is proved that under the multiplicative update rules the distance \( \|A - WH\|^F \) is monotonically non-increasing. This algorithm is a fixed-point type method. The overall cost of Algorithm 1 is \#iterations \( \times \) \( O(\text{rank}) \).

D. Block Coordinate Descent Method

By block coordinate descent method in bound-constrained optimization proposed by Bertsekas [28], we can update \( W^{k+1} \) on \( (W^k, H^k) \) and \( H^{k+1} \) on \( (W^{k+1}, H^k) \) alternatively.

Algorithm 2: Alternating Non-negative Least Squares

1. Initialize \( W_0^i > 0, H_0^j > 0, \forall i, a, b, j. \)
2. For \( k=1, 2, \ldots \)
   \[ W^{k+1}_i \in \arg \min_w f(W, H^k) \]  
   (9)
   \[ H^{k+1}_j \in \arg \min_h f(W^{k+1}, H) \]  
   (10)

Expressions (9) and (10) are sub-problems. When one block of variables is fixed, each sub-problem is indeed the collection of several non-negative least square problems. In (10), the \( j \)th column of \( H^{k+1} \) is an optimal solution of

\[ \min_{W, H} \left\| A^T(j) - W^{k+1} H \right\|^2 \]  
\[ \text{subject to } H_h \geq 0, b = 1, \ldots, k, \]  

(11)

Projected Newton’s methods are suggested to solve each problem in (11). Solving (9) and (10) per iteration could cost a lot more than the simple update in Algorithm 1. Thus efficient methods to solve sub-problems are essential.

E. Alternating Non-negative Least Squares Using Projected Gradients [29]

Lin proposes two methods for NNMF by applying projected gradient method to solve non-negative least squares problem in Algorithm 2 or directly minimize (3). Sub-problem (10) consists of \( m \) independent non-negative least squares problems (11). In this method, they are solved together rather than separately in Algorithm 2 and sub-problem (10) is rewritten as

\[ \min_H \left\| f(H) = \frac{1}{2} \|A - WH\|^2_F \right\| \]  
\[ \text{subject to } H_{bj} \geq 0, \forall b, j \]  

(12)

and an improved projected gradient method is used to solve (12).

This method leads to faster convergence than the popular multiplicative update method, and the overall cost is \#iterations \( \times (O(\text{nnr}) + \#\text{sub-iterations} \times O(\text{tnr}^2 + \text{mnr}^2)) \).

Algorithm 3: An improved projected gradient method

1. Given \( 0 < \beta < 1, 0 < \sigma < 1 \)
2. Initialize any feasible \( X^k \) and set \( \alpha_0 = 1 \).
3. For \( k=1, 2, \ldots \)
   a) Assign \( \alpha_k \leftarrow \alpha_{k-1} \)
   b) If \( \alpha_k \) satisfies
Iterative Matrix Factorization Techniques for High-Accuracy Privacy Protection on Non-negative-valued Datasets

\[ X^{k+1} = P[X^{k} - \alpha_{k} \nabla f(X^{k})] \] where \( \alpha_{k} = \beta \), and \( t_{k} \) is the first non-negative integer \( t \) for which \( f(X^{k+1}) - f(X^{k}) \leq \sigma \nabla f(X^{k})(X^{k+1} - X^{k}) \)

Then repeatedly increase it by \( \alpha_{k} \) until \( \alpha_{k} \) satisfies.

c) Set \( X^{k+1} = \lambda(\alpha_{k}) \)

F. Iteration Stopping Conditions

Without a predefined stopping condition, iterative steps will continue forever. A limitation on running time or the number of iterations is often used to interrupt the infinite loop. The difference between two recent iterations can also be a stop condition. If the difference is small enough, then the loop stops. Such a stopping condition does not reveal whether a solution is close to a stationary point or not. However, in the context of data distortion, we do not need an accurate factorization. We only require a sparse low-rank non-negative approximation of the original matrix. In our application, a requirement on data distortion level is integrated as a stopping condition to the iterative procedure. In practical, with a comprehensible consideration on data dimension, data distortion level and accuracy level, and computation cost, determining a simple limitation on norm, running time or the number of iterations by trial-and-error is an economical way.

G. Iterative NNMF Data Distortion Method

Our proposed method consists of three parts: Initialization, Iterative Loop and Distortion. Each part includes several steps detailed in Algorithm 4.

IV. EVALUATION MEASURES

The issue here to address is to hide to the outside world sensitive individual data while retaining the underlying data pattern and semantics to enable a construction of an accurate decision model on the distorted data. Therefore two categories of evaluation metrics are suggested here to perform evaluation of the new method.

A. Data distortion measures

The privacy protection measure is used to evaluate dissimilarity between the original and distorted data. It indicates how closely the original value of an item can be estimated from the distorted data. Some privacy metrics have been proposed in the literature. Some data distortion measures defined in [2] are used here to assess the level of data distortion which only depends on the original matrix \( A \) and its distorted counterpart \( A' \).

1) Value Difference (VD)

After a data matrix is distorted, the value of its elements changes. The value difference (VD) of the datasets is represented by the relative value difference in the Frobenius norm. Thus VD is the ratio of the Frobenius norm of the difference of \( A \) from \( A' \) to the Frobenius norm of \( A' \):

\[ VD = \frac{\|A - A'\|_{F}}{\|A'\|_{F}} \] (13)

2) Position Difference

After a data distortion, the order of the value of the data elements changes, too. We use several metrics to measure the position difference of the data elements. Rank Position (RP), Rank Maintenance (RM), Change of Rank of Attributes (CP), and Maintenance of Rank of Attributes (CK) are used in our experiments. Detailed definition and calculation are described in [2].
Algorithm 4 Iterative NNMF Data Distortion Method

Input:
- $A_{nm}$: non-negative matrix,
- $k$: size of dimension
- tol: limit values of errors and stopping conditions

Output:
- $r$: the final reduced dimension.
- $\tilde{A}(r)$: the final distorted dataset.

Initialization:
1. Preprocessing the original dataset $A_{nm}$
2. Examine its non-negative property;
3. Set up stopping condition: $S$
4. Set up dimension value $k < \min(n,m)$
5. Randomly generate initial estimate of non-negative matrices $(W^{(0)}_{nk}, H^{(0)}_{km})$

Iterative Loop:
6. Compute initial value of stopping condition, $S^{(0)}$
7. For each iteration $i = 0, 1, ..., $ until stopping condition satisfied, Do
8. Compute $(W^{(k+1)}_{nk}, H^{(k+1)}_{km}) = \text{NNMF} \_ \text{algorithm}(W^{(k)}_{nk}, H^{(k)}_{km})$
9. Compute $S^{(k+1)}$
10. If $S^{(k+1)}$ satisfies stopping condition,
11. Output $W$ and $H$;
12. Stop;
13. EndIf
14. EndDo

Distortion:
15. Compute approximation $\tilde{A} = WH$
16. Choose an integer $r < k$
17. For $r = k, k-1, ..., 1$, Do
18. Do further distortion: $\tilde{A}^{(r)} = W_{nr}, H_{rx}$
19. Compute privacy metrics on $\tilde{A}^{(r)}$
20. Train classifier on $\tilde{A}^{(r)}$ and compute classification accuracy.
21. EndDo
22. Choose one $\tilde{A}^{(r)}$ with satisfied privacy level and accuracy
23. Publish the final distorted dataset $\tilde{A}^{(r)}$

B. Data Utility Measure

Data utility measures indicate the accuracy of data mining algorithms on distorted data after the manipulation of certain perturbation. In this paper, Support Vector Machine (SVM) classification is chosen as the data utility measure by building a classifier on distorted dataset and applying five-fold cross validation method to compute classification accuracy as a reasonable data utility measure [30].

V. EXPERIMENTS AND RESULTS

The experiments here are designed in three steps: dataset creation, data distortion and measurement calculation. A real non-negative-value dataset is used in our experiments to examine the performance of the proposed new data distortion strategies and compare with our previous proposed strategies. All implementations of NNMF code are available at http://www.csie.ntu.edu.tw/~cjlin/nmf. Experiments are conducted on a Sunblade 150 workstation.
A. Real Dataset Descriptions

A large number of datasets from different application domains (such as financial, medical, scientific, demographic, military environments) can be used to identify the performance of our proposed approach. An example of such dataset is the Wisconsin Breast Cancer (WBC) dataset available from the UCI Machine Learning Repository at http://www.ics.uci.edu/~mlearn/MLRepository.html.

<table>
<thead>
<tr>
<th>Number</th>
<th>Attribute</th>
<th>Domain</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Sample code number</td>
<td>Id number</td>
</tr>
<tr>
<td>2</td>
<td>Clump Thickness</td>
<td>1 - 10</td>
</tr>
<tr>
<td>3</td>
<td>Uniformity of Cell Size</td>
<td>1 - 10</td>
</tr>
<tr>
<td>4</td>
<td>Uniformity of Cell Shape</td>
<td>1 - 10</td>
</tr>
<tr>
<td>5</td>
<td>Marginal Adhesion</td>
<td>1 - 10</td>
</tr>
<tr>
<td>6</td>
<td>Single Epithelial Cell Size</td>
<td>1 - 10</td>
</tr>
<tr>
<td>7</td>
<td>Bare Nuclei</td>
<td>1 - 10</td>
</tr>
<tr>
<td>8</td>
<td>Bland Chromatin</td>
<td>1 - 10</td>
</tr>
<tr>
<td>9</td>
<td>Normal Nucleoli</td>
<td>1 - 10</td>
</tr>
<tr>
<td>10</td>
<td>Mitoses</td>
<td>1 - 10</td>
</tr>
<tr>
<td>11</td>
<td>Class</td>
<td>2 for benign, 4 for malignant</td>
</tr>
</tbody>
</table>

| Class distribution: | Benign: 458 (65.5%) | Malignant: 241 |

The target WBC dataset is a 699 by 10 matrix with the 10th column representing class label.

B. Notation Description

Notations in experiments are described in Table III.

C. Default Value of Experimental Parameters

For all the experiments here, the default values of some parameters in distortion methods are listed as follows:

1. NMF: tolerance for stopping condition \( \text{tol}=1\times10^{-4} \), time limit = 4000, iteration number limit = 20000.
2. ND: the normally distributed noise is generated with \( \mu = 0 \) and \( \sigma = 0.46 \), see [2] for the meaning of these two parameters.
3. UD: the uniformly distributed noise is generated from the interval \([0, 0.8]\).
4. STS sparsification: the threshold value \( \epsilon = 0.001 \)
5. CTS sparsification: \( \epsilon = 0.2 \)
6. ETS sparsification: \( \epsilon = 0.01, \alpha = 0.2 \)
7. SVM classification: radial base function (RBF) is chosen as the kernel function and \( \gamma = 0.001 \).

D. Comparison of two iterative NMF algorithms: Experiment 1

The two NMF algorithms are implemented on WBC to compare the performance. One is multiplicative update in Algorithm 1, denoted by NMFM. The other is alternating projected gradients for each sub-problem, denoted by NMF. The problem size is \((n,k,m)=(699,7,9)\). All the tests share the same initial estimate of \((W^{(0)}_{699\times7}, H^{(0)}_{7\times9})\). The tolerance is set to be \(10^{-3}, 10^{-4}, 10^{-5} \) and \(10^{-6}\) in order to examine convergence speed. We also impose a time limit of 4000 seconds and a maximal number of 50000 iterations on each method.
Table IV shows that when the tolerance is $10^{-5}$, NMFM often exceeds the iteration limit of 50000. Obviously NMF is superior to NMFM. The data in the following experiments are collected by using NMF algorithm in iterations.

**TABLE IV** 

<table>
<thead>
<tr>
<th>Tolerance</th>
<th>Number of Iteration</th>
<th>Iteration Time (seconds)</th>
<th>Final Gradient Norm</th>
<th>Objective Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>1e-3</td>
<td>17</td>
<td>3060</td>
<td>0.8</td>
<td>1.04</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.6</td>
<td>NMFM</td>
<td>7.11</td>
</tr>
<tr>
<td>1e-4</td>
<td>94</td>
<td>20000</td>
<td>3.6</td>
<td>23.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.09</td>
<td>NMFM</td>
<td>1.54</td>
</tr>
<tr>
<td>1e-5</td>
<td>386</td>
<td>50000</td>
<td>9.8</td>
<td>49.7</td>
</tr>
<tr>
<td></td>
<td></td>
<td>0.01</td>
<td>NMFM</td>
<td>0.84</td>
</tr>
<tr>
<td>1e-6</td>
<td>2382</td>
<td>-</td>
<td>63.3</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NMFM</td>
<td>41.4</td>
</tr>
</tbody>
</table>

Initial objective value: 276.2; Initial Gradient Norm: 7609.7; dimension:7;. When tolerance is greater than 1e-5, number of iteration of NMFM exceeds the prescribed limit.

**E. Performance of NMF Algorithm using Projected Gradients: Experiment 2**

An initial random guess on $W$ and $H$ is the first step in the beginning of iteration. Different start value leads to different initial gradient norm. Therefore, the result and iteration time are dependent on the initial guess. The computation cost are roughly examined on dimension value from 9 to 2 under the tolerance value 1e-4.

**TABLE V** 

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Initial Gradient Norm</th>
<th># of Iterations</th>
<th>Iteration Time(seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>16525</td>
<td>83</td>
<td>12.41</td>
</tr>
<tr>
<td>8</td>
<td>11584</td>
<td>94</td>
<td>7.44</td>
</tr>
<tr>
<td>7</td>
<td>10648</td>
<td>80</td>
<td>7.38</td>
</tr>
<tr>
<td>6</td>
<td>7499</td>
<td>109</td>
<td>8.84</td>
</tr>
<tr>
<td>5</td>
<td>4816</td>
<td>117</td>
<td>7.85</td>
</tr>
<tr>
<td>4</td>
<td>5196</td>
<td>128</td>
<td>9.2</td>
</tr>
<tr>
<td>3</td>
<td>3265</td>
<td>76</td>
<td>4.65</td>
</tr>
<tr>
<td>2</td>
<td>4312</td>
<td>20</td>
<td>0.52</td>
</tr>
</tbody>
</table>

**F. Sparseness Level of $W$ and $H$: Experiment 3**

NNMF factorization makes two submatrices with higher sparseness than those by singular value decomposition. In the experiment, the sparseness of a vector $x$ of length $n$ is defined as

$$\text{sparseness}(x) = \sqrt{n - \|\|} / \sqrt{n - 1}$$

To measure sparseness of a matrix, we stack columns of the matrix to form a vector. The maximal of sparseness of $x$ is 1 if containing $n-1$ zeros, and it reaches zero if the absolute value of all coefficient of $x$ coincide.

Fig.1. and Fig.2. illustrate the bar plots of $W$ and $H$ created by NMF algorithm on WBC with k=7 and tolerance=$10^{-4}$. The sparseness of $W$ and $H$ are 0.34 and 0.64 respectively. More than 50% of entries in $H$ are zeros. The algorithms to solve $W$ and $H$ used in our method make $H$ more sparse than $W$. Hence, in the natural interpretation of the factorization, $H$ is the basis or factor vectors and it tends to be sparse. Implicitly this suggests that the basis will involve only some of the original attributes. While that $W$ is denser than $H$ implies the objects are combinations of all basis.

**G. Comparison of Iterative NNMF Data Distortion Strategies with SVD, UD and ND on WBC: Experiment 4**

The ten distortion methods, NNMF-based, uniformly distributed noise (UD), normally distributed noise (ND), SVD, SSVD, SSVD with matrix partition, are implemented on WBC to compare the performance. In order to be fair in comparing the data distortion metrics, parameters are set to such certain values as to make VD values of UD, ND, SVD and NMF as close as possible. Rank $\kappa$ of SVD is 7. Dimension size in NMF is 7 and final dimension is also 7. The results of performance evaluation on the ten methods are provided in Table VI and Fig.3.
Iterative Matrix Factorization Techniques for High-Accuracy Privacy Protection on Non-negative-valued Datasets

Fig. 1. Sparseness of Basis / Factor Vectors $H$, created by NMF algorithm on WBC with $k=7$ and tolerance=$10^{-4}$. Sparseness of $H$ is 0.64.

Fig. 2. Sparseness of Object Vectors $W$, created by NMF algorithm on WBC with $k=7$ and tolerance=$10^{-4}$. Sparseness of $W$ is 0.34.

<table>
<thead>
<tr>
<th>Methods</th>
<th>VD</th>
<th>RP</th>
<th>RK</th>
<th>CP</th>
<th>CK (classification)</th>
<th>Accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>WBC</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>UD</td>
<td>0.1085</td>
<td>219.6993</td>
<td>0.0130</td>
<td>0</td>
<td>1</td>
<td>96.4</td>
</tr>
<tr>
<td>ND</td>
<td>0.1098</td>
<td>224.8148</td>
<td>0.0084</td>
<td>0</td>
<td>1</td>
<td>96.3</td>
</tr>
<tr>
<td>SVD</td>
<td>0.1222</td>
<td>228.8972</td>
<td>0.0114</td>
<td>0.2222</td>
<td>0.7778</td>
<td>96.4</td>
</tr>
<tr>
<td>NMF</td>
<td><strong>0.1228</strong></td>
<td><strong>228.4295</strong></td>
<td><strong>0.0100</strong></td>
<td><strong>0.2222</strong></td>
<td><strong>0.7778</strong></td>
<td><strong>96.7</strong></td>
</tr>
<tr>
<td>SSVD</td>
<td>1.2662</td>
<td>228.1370</td>
<td>0.0013</td>
<td>3.3333</td>
<td>0</td>
<td>96.6</td>
</tr>
<tr>
<td>CSVD</td>
<td>1.2702</td>
<td>230.1561</td>
<td>0.0021</td>
<td>3.3333</td>
<td>0</td>
<td>96.4</td>
</tr>
<tr>
<td>ESVD</td>
<td>1.2704</td>
<td>228.0744</td>
<td>0.0014</td>
<td>3.3333</td>
<td>0</td>
<td>96.4</td>
</tr>
<tr>
<td>SNMF</td>
<td>0.1228</td>
<td>228.4362</td>
<td>0.0076</td>
<td>0.2222</td>
<td>0.7778</td>
<td>96.4</td>
</tr>
<tr>
<td>CNMF</td>
<td>0.1297</td>
<td>226.5042</td>
<td>0.0081</td>
<td>0.2222</td>
<td>0.7778</td>
<td>96.5</td>
</tr>
<tr>
<td>ENMF</td>
<td>0.1234</td>
<td>228.2035</td>
<td>0.0089</td>
<td>1.1111</td>
<td>0.5556</td>
<td>96.5</td>
</tr>
</tbody>
</table>

Note: Parameters: SVD: $k=7$, NMF: $k=r=7$. The value of VD is adjusted as close as possible for UD, ND, SVD and NMF, in order to make a fair comparison.

Under the premise on the same level of value dissimilarity, the fact that CP value of UD and ND is 0 and CK value be 1 indicates that additive noise methods are worse than matrix-decomposition-based methods.

Experimental data in Table VI supports the following conclusions:

1. NNMF-based distortion strategies achieve a comparable performance with SVD-based strategies. In particular, it achieves the highest classification accuracy.
2. No improvement on performance of NNMF-based methods by applying the sparsification strategies. It is reasonable under the condition that NNMF is a sparse factorization and two factors, $W$ and $H$, have a deep level of sparseness. Thus, further sparseification does not provide any improvement.
3. Sparse SVD performs best on data distortion level without any degradation on data mining accuracy. It is obvious that sparsification has a strong effect on data distortion level of SVD-based methods by making all the attributes change their rank in average value because CK value is 0.
4. As to data utility, all the ten methods achieve a level at least not worse than using the original dataset.

H. Sensibility of Performance on Dimension of NNMF: Experiment 5

To examine the effect of dimension size on privacy level and data utility level in the approximation, we conduct the experiment on WBC and Fig.2 illustrates the influence of dimension size on data distortion level and classification accuracy. Here $W$ and $H$ are solved under the dimension of 7. Then the final compressed approximation of WBC is computed by setting up $r$ from 6 to 2.

Dimension size is a key element both for dimension reduction and data distortion level. The smaller the
dimension size is, the higher the data distortion level of the method is. However, clearly, the dimension size is inversely proportional to data utility level. Fig. 4 illustrates the above relationship. How to choose the optimal dimension size in the proposed method is an empirical problem. For WBC, our experiments imply one possible good choice for our distortion method considering both data utility and data privacy. When the initial dimension size is 7, we can choose 4 as a reasonable size.

I. Summary and Remarks

In this paper, we considered high accuracy data distortion of non-negative-valued datasets. The important property of non-negative matrix factorization, non-negative and sparseness, makes it not only a good dimension reduction technique but also an efficient privacy preserving tool. Experimental results indicate that by a careful choice of iterative parameter settings, two sparse non-negative factors can be computed by some efficient iterative algorithms. Alternating least squares using projected gradients in computing NNMF converges faster than multiplicative update methods. The value of these two matrices is not unique because it is dependent on initial estimates in the beginning of iterative procedure. This dependency provides our method both with uncertainty and flexibility. For non-negative-valued datasets, our proposed method provides a possibility of simultaneously achieving satisfactory privacy, accuracy and efficiency. With the same level of distortion as the other data distortion methods, the NMF method demonstrates the highest classification accuracy. In particular, we foresee that using iterative factorization of the original datasets, there is an opportunity where all these three goals can be reached.

ACKNOWLEDGMENT

This research work of the authors was supported in part by the U.S National Science Foundation under grants CCR-0092532 and CCF-0527967, in part by the Kentucky Science and Engineering Foundation under grant KSEF-148-502-05-132, and in part by the Alzheimer’s Association under a New Investigator Research Grant NIGR-06-25460.

REFERENCES

Iterative Matrix Factorization Techniques for High-Accuracy Privacy Protection on Non-negative-valued Datasets


[23] D. Guillamet and J. Vitria, “Determining a suitable metric when using non-negative matrix factorization,” 16th International Conference on Pattern Recognition (ICPR’02), vol.2, Quebec City, QC, Canada, 2002

