Feature Selection: A Preprocess for Data Perturbation

Pengpeng Lin, Nirmal Thapa, Ingrid St. Omer, Lian Liu and Jun Zhang

***Abstract*—As a major concern in designing various data mining applications, privacy preservation has become a critical component seeking a trade-off between mining performances and protecting sensitive information. Data perturbation or distortion is a widely used approach for privacy protection. Many privacy preservation approaches were developed, either by adding noises or by matrix decomposition methods. In this paper, we intensively studied Singular Value Decomposition (SVD) based data distortion strategy and feature selection techniques, and conducted experiments to explore how feature selection technique could be used and better serve for privacy preservation purpose. Sparsified Singular Value Decomposition (SSVD) and filter based feature selection are used for data distortion and reducing feature space. We design a modified version of Exponential Threshold Strategy (ETS) as our threshold function for matrix sparsification process, and implement several metrics to measure data perturbation level. We also propose a novel algorithm to compute rank and analyze its lower running time bound. The mining utility of distorted data is tested with a well known Classifier, Support Vector Machine (SVM).**

***Index Terms*—SVD; SSVD; SVM; feature selection; perturbation**

I. Introduction

PRIVACY preserving data mining (PPDM) and privacy preserving data publishing (PPDP) are two closely related research directions. The former concentrates on privacy issues when data miners requesting real data for the mining purpose; the latter stresses on an application-free protection of data whenever in need of publishing data for business transactions or research purpose. Both of them disguise dataset in an effort to replace the original dataset for data publications and data mining applications. With the rapid growth of data exchange technology, collaborations with information between different parties become essential approach in many situations for business and research activities. Without an acceptable level of privacy of sensitive information, many data mining applications would not be applicable. How can an entity be entrusted with access to sensitive personal or business information, and how can sensitive datasets be sufficiently protected from unauthorized access without undermining accuracy of mining knowledge are the important issues. Data privacy preservation is premised on the maintenance of data analytical values. Preserving privacy of data sets while still being able to extract valid data mining results is a very challenging task. Among the widely used approaches, Singular Value Decomposition (SVD) is one of the most popular techniques to the above addressed issues. Its derivative, Sparsified Singular Value Decomposition (SSVD) concept was firstly introduced by Gao and Zhang in [2] for reducing the storage cost and enhancing the performance of SVD in text retrieval applications. Xu et al. applied SVD and SSVD methods in a terrorist analysis system [3]. SSVD was further studied in [4] in which matrix structural partition strategies were proposed and used to partition the original data matrix into submatrices. The computational cost incurred by matrix decomposition phase is substantially reduced. In [5], Wang suggested that significance of features for analysis purposes should be taken into consideration and all features were ranked by using feature selection methods. The objective of feature selection is to select most correlated features regarding mining target while eliminating the unrelated data and reducing dataset dimensionality and hence, saving computational expense and achieving better accuracy of mining results. However, the questions are that can analysis results of data be preserved by performing data distortion technique on selected features using feature selection methods? And how can feature selection methods produce better result or result in tolerable error rate on perturbed data? Is it better to perform feature selection before data distortion or is it better the other way around? In our work, we take a close look at these interesting questions. Mainly, three experiments are conducted in our work to answer the questions above. We select subfeature set according to their significance ranked by using filter based feature selection method. The selected subset is then distorted by using SVD modification approaches. We carry out experiments by interchanging the sequence of feature selection and SVD data distortion procedure. The Support Vector Machine (SVM) and several distortion metrics are used in the experiments to measure for data mining quality and data distortion level respectively.

The remainder of the paper is organized as follows. Sect.II briefly introduces related knowledge such as privacy preserving data mining, essential SVD and SSVD processes, feature selection methods, and SVM method. Sect.III discusses various data distortion metrics, their usages and we propose a novel algorithm to compute rank and estimate its run time complexity. The experiments are carried out and the results are presented and discussed in Sect.IV. We finally sum up this paper and bring our future plans in Sect.V.

Manuscript received April 29, 2011; revised May 10, 2011.

P.Lin is with Department of Computer Science, University of Kentucky, Lexington, KY, 40506-0046 USA. e-mail: M.Lin@uky.edu. P.Lin would like to thank the Kentucky-West Virginia Alliance for Minority Participation Program (NSF Award #0603091) for supporting his graduate research

Nirmal Thapa is with Department of Computer Science, University of Kentucky, Lexington, KY, 40506-0046 USA. e-mail: nirmalthapa@uky.edu

Ingrid St. Omer is an Assistant Professor in the Department of Electrical and Computer Engineering, University of Kentucky, Lexington, KY, 40506-0046 USA. e-mail: istomer@engr.uky.edu

L.Liu is with Department of Computer Science, University of Kentucky, Lexington, KY, 40506-0046 USA. e-mail: lian.liu@uky.edu

J.Zhang is a Professor in the Department of Computer Science, University of Kentucky, Lexington, KY, 40506-0046 USA. e-mail: jzhang@cs.uky.edu

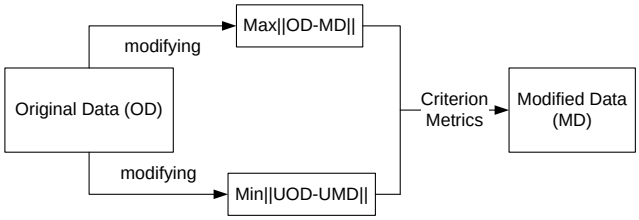

Fig. 1. PPDM, UOD and UMD denote for utility of original data and Modified data, respectively

II. Background and Related Work

A. Privacy Preserving Data Mining

Data mining is a recently growing field which merges the knowledge of databases, statistics and machine learning together. It is also known as knowledge discovery and attempts to extract meaningful information and useful patterns from raw data. As the data mining techniques gaining popularity and widely being used in business and research, there has been a raising concern for disclosure of security and privacy. The recent advances in data collection and dissemination through Internet or other media have made threats against the privacy very common on a daily basis [11]. For example, two parties each having a private databases with sensitive contents, wish to work in collaboration by applying a data mining algorithm on the union of their databases. Indeed, neither party wants their private data to be known to other party. In such scenario, Privacy preserving data mining (PPDM) can be used to hide sensitive information. In general, the main consideration in PPDM is two folds: hiding sensitive features such as age, salary, or personal medical conditions and sensitive knowledge that can be discovered by data mining algorithms. PPDM develops algorithms for modifying the original data in some way that the private data and private knowledge remain private even after the mining process [12]. Common techniques include data perturbation, blocking feature values, swapping tuples, or merging feature values into an aggregated and coarser granularity, etc. However, It is also important to realize that modifying data will results in degradation of the data utility. As illustrated in the Figure 1, a PPDM scheme should be able to maximize the degree of data modification, while retain the maximum data utility level. In the next sub-section, we briefly discuss the well-known SVD matrix decomposition technique that has been used in [3,4] to achieve this objective.

B. Singular Value Decomposition

Without loss of generality, we let A be a matrix in $R^{m\times n}$ with $m \geq n$ (the following results in this paper also hold for the assumption that $m \leq n$). Let r denote the rank of A, where $r \leq n$. We use lower case letters for scalar, lower case letters with under bar for vector, and capital letters for matrix, e.g., A is a matrix, a is a scalar, $\underline{a}$ is a column vector, and the row vector is denoted as $\underline{a}^T$. Those notations will be used throughout this paper.

- *Def:* Any matrix $A \in R^{m\times n}$ can be decomposed uniquely as:

$$A = UDV^T \tag{1}$$

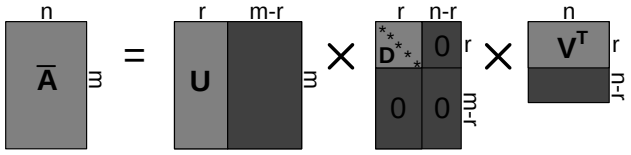

Fig. 2. Singular Value Decomposition

Where U is $m\times m$ orthonormal matrix, V is $n\times n$ orthonormal matrix. D is $m\times n$ diagonal matrix whose non-negative entries on its diagonal are called singular values. This is called singular value decomposition method and can be proved constructively. The interested readers can refer to [13] for details.

SVD has various important mathematic properties. Let $\delta(\sigma_1, \sigma_1, \ldots, \sigma_k) = diag(D)$, where $k = min(m, n)$, the singular values are ordered such that $\sigma_1 \geq \sigma_2, \ldots, \geq \sigma_k$. Correspondingly, we have $\lambda_i \subseteq (\sigma_1^2, \sigma_2^2, \ldots, \sigma_k^2)$, where $i = 1, \ldots, k$ and λ_i represents the egienvalues of $A^T A$. Let $\underline{x}_i$ be the eigenvector belonging to λ_i. It follows that:

$$\|A\underline{x}_i\|^2 = \underline{x}_i^T A^T A \underline{x}_i = \lambda_i \underline{x}_i^T \underline{x}_i = \lambda_i \|\underline{x}_i\|^2 \tag{2}$$

Hence

$$\lambda_i = \frac{\|A\underline{x}_i\|^2}{\|\underline{x}_i\|^2} \tag{3}$$

The equation (3) indicates that the induced operator two norm of A equals σ_1, the largest singular value. Since the rank of A equals the number of singular values, it further implicates that the main characteristics of A can be captured by lower rank items. On the other hand, the singular values around the bottom of the diagonal of D are relatively small and can be considered insignificant. If we introduce perturbations on those insignificant singular values i.e., making them zero, we can represent A in a perturbed form $\bar{A}$. In addition, the removed part E, where $E = A - \bar{A}$, can be considered as noise in A [7]. Thus, $\bar{A}$ can be seen as both a distorted copy of A and a faithful representation of the original data [4].

SVD also has numerous applications in data mining, information retrieval and image compression in which it is often used to approximate a given matrix by a lower rank matrix with the minimum distance between them. As an example, the space storage for the $m \times n$ data A, would require much less with SVD technique, i.e., A requires $m\times n$ storage, whereas the three decomposed matrices U, D and V only requires $m \times r + r \times r + r \times n$ storage (see Figure 2 for reference), where r is much smaller than n. Generally speaking, SVD technique is often chosen to determine a matrix approximation with smaller rank.

C. Sparsified SVD

The sparsification process of a matrix A is to set a threshold and the entry values of A less than the threshold are zeroed out. We apply this strategy to the decomposed matrix D to get perturbed diagonal matrix $\widetilde{D}$. Essentially, we keep k largest singular values and set the rest zero. Since the number of singular value equals the rank of the matrix, it can be seen from Figure 2 that a distorted matrix $\bar{A}$ of low rank can be composed by simple block matrix operations:

$$\bar{A} = UDV^T \tag{4}$$

To increase distortion level, we also perform sparsification process on U and V. The sparsification operation is referred to as dropping operation in [2]. We then multiply all three sparsified matrices $\widetilde{U}$, $\widetilde{D}$ and $\widetilde{V}$ to get $\widetilde{A}$:

$$\widetilde{A} = \widetilde{U}\widetilde{D}\widetilde{V}^T$$

If we denote S the sparsification function for U and V, then it is easy to see that:

$$\widetilde{A} = S(\overline{A})$$

And since after sparfication process, many small column vectors in U and V are dropped to zero, then we have:

$$\begin{aligned}\widetilde{A} &= S(A_1) + S(A_2)+,\ldots,+S(A_k)\\ &= S(\underline{u}_1\sigma_1\underline{v}_1^T) + S(\underline{u}_2\sigma_2\underline{v}_2^T)+,\ldots,+S(\underline{u}_k\sigma_k\underline{v}_k^T)\\ &= \sigma_1 S(\underline{u}_1)S(\underline{v}_1^T) + \sigma_2 S(\underline{u}_2)S(\underline{v}_2^T)+,\ldots,+\sigma_k S(\underline{u}_k)S(\underline{v}_k^T)\\ &= S(A_1)+,\ldots,+0 + S(A_j)+,\ldots,+0+,\ldots,+S(A_k)\end{aligned}$$

Therefore, the $\widetilde{A}$ by SSVD process can be seen as a matrix from further perturbing $\overline{A}$:

$$A = \widetilde{A} + E_1 + E_2 \tag{5}$$

Where $E_1 = A - \overline{A}$ and $E_2 = \overline{A} - \widetilde{A}$. After operating sparsification process on U and V, the significant values are still kept, thus the mining utility of *A* is well preserved and its entry values are distorted twice at the same time[4].

D. Sparsification Strategy

Three *sparsification* strategies were proposed in [2], where the Exponential Threshold Strategy (ETS) showed the best empirical results. In our work, we design a modified ETS threshold function named *METS*. METS, as in (6), defines a smooth threshold function using an exponential function in which the threshold value is customized for each column of the matrix.

$$T_j = \frac{\epsilon}{m}\sum_{i=1}^{m} |a_{ij}|\, e^{j\cdot r^{-2}} \tag{6}$$

The original ETS threshold formula is modified in METS by having parameter α redefined. Rather than setting different value for α every time, we substitute it with a fraction number r^{-2}, whose magnitude is determined by r, which is the number of the singular values kept. The computed threshold value for each column is adjustable with scaling factor ϵ. Note that different from ETS, the absolute value of a_{ij} is computed in METS. This is because that during SVD decomposition, some of the entries in decomposed matrices U and V might be negative. As a result, the threshold calculated based on the original ETS formula may be large for low rank items and small for high rank items. Calculating threshold value with absolute entry value ensures that larger threshold values are computed for entry value with higher column index. Therefore, the most important entries are kept, whereas more trivial entries will be zero[2].

E. Feature Selection

Feature selection research has found applications in many fields where large volumes of data present challenges to effective data analysis and processing. As data evolves to be ubiquitous and abundant, new challenges arise everyday and expectations of feature selection are also elevated. Feature selection algorithms have two main components: *feature search* and *feature subset evaluation*.

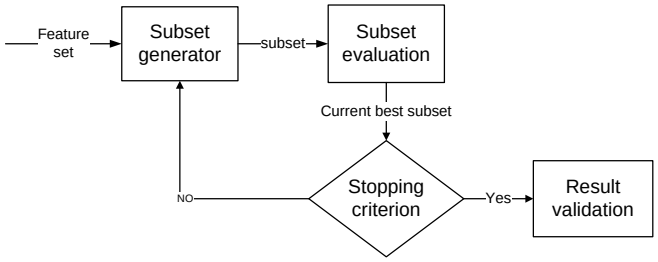

Fig. 3. Feature Selection Process

Feature search strategies have been widely used for searching feature space. An exhaustive search would certainly find the optimal solution; however, for a dataset of *N* features, a search on 2^N possible feature combinations is obviously computationally impractical. More realistic search strategies have been studied to make the problems more tractable. Sequential search methods generally use greedy approach and result in an $O(N^2)$ worst case search. Marill and Green [5] proposed the sequential backward selection, which starts with full feature space and sequentially eliminates the feature that contributes least to the criterion function one at a time. Whitney [6] introduced sequential forward selection, which starts with empty set and sequentially adds one feature at a time. Random search methods such as genetic algorithms add some randomness in the search procedure to escape from a local optimum. Individual search methods evaluate each feature individually and select features that either satisfy the condition or are top-ranked. In our work, a sequential search *Best First Search* (BFS) is used in the experiments.

Feature subset evaluation process as in Figure 3 is used to identify irrelevant and redundant features. In classification, the feature evaluation criterions are naturally related to the labeled classes, thus filter based methods are often used. In clustering where class labels may be unavailable, either filter or wrapper approaches are used. As shown in Figure 4, the wrapper approach wraps the feature search with learning algorithms whereas filter approach utilizes the intrinsic property of the data alone to select feature subspace. Intuitively, wrapper approach may result in a better performance. However, wrapper methods are more expensive since they run the learning algorithm for each candidate feature subset. In our experiments, we use filter method selecting features for data classification and employ support vector machine (SVM) for the data utility measurement.

F. Support Vector Machine

Support Vector Machine (SVM) is chosen in our work as the data utility metric to assess how much a dataset keeps the analytical values of data mining techniques after a data perturbation process. SVM is a method for classification. It uses a nonlinear mapping to transform the original training data that are linearly inseparable into a higher dimension. It then searches for the linear optimal separating *hyperplane*. A *hyperplane* that separates data from different classes can

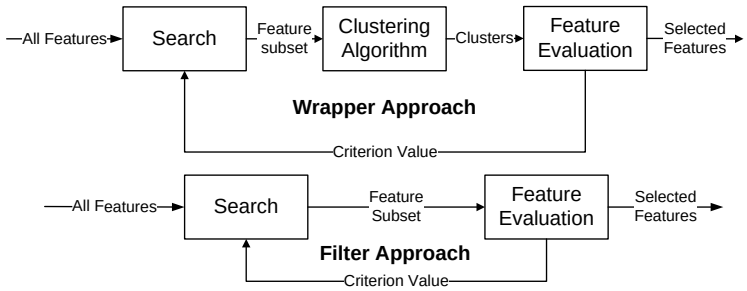

Fig. 4. Filter and Wrapper Approaches

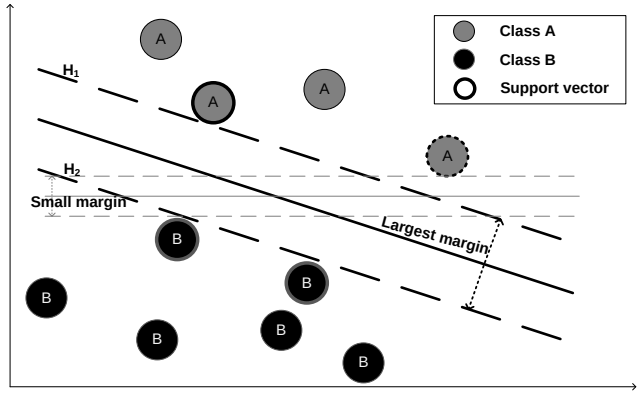

Fig. 5. Support Vector Machine

always be found by mapping data into a sufficiently high dimension. The basic SVM process is shown in Figure 5. Essentially, two *hyperplanes* H1, H2 with maximum *margin* are defined for every class pairs. Any training tuples that fall on H1 or H2 are called *support vectors*. Tuples that falls on or above H1 belong to class A, and tuples that falls on or below H2 belong to class B. The SVM finds the *hyperplane* using *support vectors* and maximum *margins* (see Figure 5 for reference).

III. Data Distortion Measurements

Data distortion metrics are used to measure the degree of data distortion. In this paper, we implemented the metrics that were introduced in the literatures [3,4]. These metrics are designed to assess the difference between original data *A* and its perturbed counterpart $\widetilde{A}$ in terms of ***Value Difference*** and ***Rank Difference***.

A. Value Difference(VD)

After a data distortion process, the value changes between original data and perturbed data is measured by (7), where Frobenius norm is used to quantify matrix $A \in R^{m \times n}$ to R.

$$VD = \frac{\left\| A - \widetilde{A} \right\|_F}{\|A\|_F} \tag{7}$$

B. Rank Difference(RD)

To measure data position changes, the values in each column are ranked in an ascending order. The ranks change between original data and perturbed data after distortion. *Rank Position* (RP) and *Rank Maintenance* (RM) [3,4] are used to measure the average change of rank for all the data values and the percentage of elements that keep their ranks of magnitude in each column after the distortion respectively [3].

One may infer the content of one feature from its relative value difference compared with the other attributes. Thus it is desirable that the order of the average value of each attribute varies after the data distortion [4]. The rank of a feature is assigned according to its average value. *Change of Rank of Features* (CP) and *Maintenance of Rank of Features* (CK) [3,4] are used in our work to indicate the changes of rank of the average value of the features and assess the percentage of the features that keep their ranks after the distortion. Table I listed all the metric formulas. Interested readers might refer to [3,4] for a detailed description.

C. Compute Ranks(CR)

We now propose a novel algorithm (CRK) to compute ranks, as shown below.

Algorithm 1 Compute Ranks (CRK)

Require: $m \times n$ DataSet *S*, $A[m][n][3]$
Ensure: *Numerical Data Type*

```
for i = 1 to n do
    for j = 1 to m do
        A(j,i)[1] ← S(j,i)
        A(j,i)[2] ← j
    end for
end for
Sort Col(A) by A(,n)[1]
for i = 1 to n do
    for j = 1 to m do
        A(j,i,3) ← j
    end for
end for
Sort Col(A) by A(,n)[2]
return A(,)[3]
```

In the ***Algorithm 1***, *A* is a multidimensional array, and each cell can hold up to 3 values. We use notation $A(m,n)[x]$ to represent each value in *A*. For example, $A(i,j)[k]$ denotes for the k^{th} value of the entry in i^{th} row and j^{th} column, where $k \in [1,3]$. Similarly, $S(i,j)$ denotes the data entry in i^{th} row and j^{th} column of *S*. If m and n are not specified, the whole row or the whole column is being considered. For example, $A(,j)[k]$ denotes for the k^{th} value in j^{th} column

TABLE I
LIST OF DATA PERTURBATION METRICS

Metric Formula	Parameter Description
$VD = \frac{\|A-\bar{A}\|_F}{\|A\|_F}$	where $A \in R^{m\times n}$
$RP = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}\left\|Rank_j^i-(Rank_j^i)^*\right\|}{m\times n}$	$(Rank_j^i)^*$ is the rank for perturbed data
$RM = \frac{\sum_{i=1}^{m}\sum_{j=1}^{n}RK_j^i}{m\times n}$	$RK_j^i = \begin{cases} 1 & \text{if } Rank_j^i = (Rank_j^i)^* \\ 0 & \text{otherwise} \end{cases}$
$CP = \frac{\sum_{i=1}^{m}\|RAV_i-(RAV_i)^*\|}{m}$	RAV_i is the rank of the average value of feature i
$CK = \frac{\sum_{i=1}^{m}CK^i}{m}$	$CK^i = \begin{cases} 1 & \text{if } RAVi = (RAV_i)^* \\ 0 & \text{otherwise} \end{cases}$

and $A(,n)[k]$ denotes for the k^{th} value of each entry in all the columns.

In the steps 1-6 of the Algorithm, the 1^{st} and 2^{nd} values of entry in *A* are assigned with the data values in *S* and their corresponding row index respectively. We then sort each column of *A* in ascending order by the first value in each entry in step 7. In the steps 8-12 we assign the 3^{rd} value of each entry in *A* with the current corresponding row index. Finally, we sort each column of *A* in ascending order by the second value in corresponding entry in step13. Step13 is to rearrange *A* back to the original form. The 3^{rd} values in a newly arranged order after step13 form a nice rank table.

We also define that if two elements in the data table have the same value, the element with the lower row index to have the higher rank. Assuming that the data set is an $n\times n$ square matrix, since comparison based sorting algorithms have lower bound o($nlog(n)$) and CRK sorts the data twice for each column, the estimated time is o($2n^2log(n)$). Since it is not growing exponentially, for a large scale data set, this is an acceptable computational cost.

IV. EXPERIMENTS AND RESULTS

We conduct experiments to test the performance of the SVM on distorted data produced by feature selection and data perturbation procedure in different sequence. The results are compared with outcomes produced by performing SVM on original data without any distortion. The perturbation sequence that generates closer result to the result produced from original data without perturbation is considered preserving better mining utility. The data distortion level and degree of feature selection are measured and compared with metrics discussed in Sect.III.

A. Setup and Dataset

We implemented threshold function METS for matrix sparsification, data distortion metrics described in [3,4] and the SVD matrix decomposition process. In order to provide a comprehensive, adequate and convincing empirical result, we conducted experiments on three real data sets and one synthetic data set. We download "Wisconsin Breast Cancer (Diagnostic)" data set, Connectionist Bench (Sonar, Mines vs. Rocks) data set and Ionosphere Radar data set from [8,9,14]. The Wisconsin Breast Cancer data set has 32 features, such as diagnosis, texture, smoothness, concavity, concave points, fractal dimension, etc. These features are computed from a digitized image of a fine needle aspirate (*FNA*) of a breast mass. They describe characteristics of the cell nuclei present in the image. The target feature is Diagnosis: "B" = benign, "M" = malignant. The dimension of the data matrix is 569×32. Connectionist Bench data set has 60 features and 208 instances. This data set contains patterns obtained by bouncing sonar signals off a metal cylinder or rocks at various angles and under different conditions. Each pattern is a set of 60 numbers in the range from 0.0 to 1.0, which represents the energy within a particular frequency band integrated over a certain period of time. For the target feature, the label associated with each record is letter "R" if the object is rock and "M" if it is a metal cylinder. The Ionosphere radar data was collected by a system in Goose Bay, Labrador. This system consists of a phased array of 16 high-frequency antennas with a total transmitted power on the order of 6.4 kilowatts. The targets were free electrons in the ionosphere. "Good" radar returns are those showing evidence of some type of structure in the ionosphere. "Bad" returns are those that do not, i.e., their signals failed to pass through the ionosphere. For the synthetic data, we use a data generator that produces data randomly by producing a decision list. The decision list consists of rules. Instances are generated randomly one by one. If decision list fails to classify the current instance, a new rule according to this current instance is generated and added to the decision list. A new rule is rejected and discarded when the total number of rules reached the maximum parameter setting.

Correlation-based feature evaluator is used to assess the worthiness of a feature subset by considering the individual predictive ability of each feature along with the degree of redundancy between them. We choose Best First Search (BFS) to search the feature space by greedy hill climbing either augmented with a forward tracking facility or decremented with a backward searching facility.

B. Experiment 1

In experiment 1, we perform feature selection (*Fs*) on original data (*Org*) without any data distortion. We then use SVM to generate the correct predict rate. Ten folds cross validation is set to split the data in 10 approximately equal parts $D_1, \ldots, D_{10}$. Training set D_i^t is obtained by removing part of D_i from D.

The results are shown in Table II, 1^{st} and 2^{nd} rows. The SVM correct predict rate for both *Wisconsin Breast Cancer* (WBC) data set and Ionosphere Radar data set with

TABLE II
SVM RESULTS

DataSet:	WBC		Sonar		Ionosphere		Synthetic Data	
exp#	**F Size**	**SVM Rate**	**F Size**	**SVM Rate**	**F Size**	**SVM Rate**	**F Size**	**SVM Rate**
Org	32	97.89%	60	75.96%	33	88.61%	36	78.17%
Fs	12	96.66%	19	77.40%	14	87.75%	5	78.17%
$Ep2$	12	92.26%	19	76.44%	14	85.76%	5	78.17%
$Ep3$	7	90.86%	13	75.00%	15	86.32%	4	78.17%

reduced feature space dropped slightly after performing feature selection method, whereas the same correct predict rate is generated for synthetic data before and after feature selection. For "Sonar" data set, 16 out of 60 features were selected and the correct predict rate, on the contrary, increased by 1.44 percent. This is due to the fact that those irrelevant features which can be regarded as noise are singled out and discarded with feature selection process. We also observe that the feature space is reduced significantly for both data sets after feature selection with only insignificant effects on correct predict rate, which indicates that both data sets consist of large proportion of unwanted information that has very little perturbation values. Applying data distortion procedure on selected feature space can result in better performance.

C. Experiment 2

In experiment 2, we carried out the experiment in the sequence that performing feature selection before data distortion. After feature selection, the data consist of reduced feature space that is most relevant to the target class. The selected features are considered to have high data perturbation value in comparison to the discarded features. We treat data as a matrix and perform SVD matrix decomposition on it. We then sparsify the three decomposed matrices U, D and V. For each singular values σ_i on the diagonal of decomposed matrix D, we define the sparsification rule as follows:

$$\sigma_i = \begin{cases} \sigma_i & \text{if } \sigma_i > 1 \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

Only the singular values greater than one are kept. In the case where $\sigma_i > 1$ for all $i \in [1, r]$, we define the sparsification rule as:

$$\sigma_i = \begin{cases} \sigma_i & \text{if } \sigma_i > \frac{3}{5r}\sum_{j=1}^{r}\sigma_j \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

Note that the sparsification rule (9) states that if all singular values are greater than zero, then we keep approximately $\lceil 7/10 \rceil$ of the total number of singular values, i.e., for a decomposed matrix D with 100 singular values, approximately 70 largest singular values are kept. Furthermore, since the number of singular values is equal to r which is the rank of the matrix A, the recomposed matrix $\overline{A}$ will be rank of about $\lceil 7r/10 \rceil$. For the decomposed matrices U and V, we use MEST to compute threshold value ζ for each column. The scaling parameter ϵ of METS is set to be 0.6. The entry values in U and V less than ζ are set to zero, or remain untouched otherwise. To be consistent, all data sets are perturbed using the same parameter settings. After sparsification, a perturbed data matrix is recomposed by multiplications of the sparsified matrices $\widetilde{U}$, $\widetilde{D}$ and $\widetilde{V}^T$. We then assess its distortion levels with the distortion metrics discussed in Sect. III. The data distortion level results are shown in Table III, Table IV, Table V, and Table VI, where *NSV* stands for number of singular values, and *SK* stands for number of singular values kept after sparsification.

TABLE III
WISCONSIN BREAST CANCER DATA

exp#	*Level Of Distortion*						
	VD	**RP**	**RM**	**CP**	**CK**	**SK**	**NSV**
$Ep2$	0.03	140.5	0.022	2.0	0.33	7	12
$Ep3$	0.33	84.95	0.015	0.0	1.0	15	31

TABLE IV
SONAR DATA

exp#	*Level Of Distortion*						
	VD	**RP**	**RM**	**CP**	**CK**	**SK**	**NSV**
$Ep2$	0.20	32.49	0.022	1.263	0.631	7	19
$Ep3$	0.18	19.63	0.033	0.308	0.769	22	60

TABLE V
IONOSPHERE DATA

exp#	*Level Of Distortion*						
	VD	**RP**	**RM**	**CP**	**CK**	**SK**	**NSV**
$Ep2$	0.28	37.31	0.018	0.29	0.71	11	14
$Ep3$	0.19	25.58	0.053	0.65	0.5	33	34

TABLE VI
SYNTHETIC DATA

exp#	*Level Of Distortion*						
	VD	**RP**	**RM**	**CP**	**CK**	**SK**	**NSV**
$Ep2$	0.18	39.81	0.011	0.8	0.4	5	5
$Ep3$	0.13	34.63	0.009	1.2	0.4	35	36

We can see from results that VD and RP values for all four data sets appear to be small due to the small data entry values. The RM values and CK values, on the other hand, explicitly indicate that these data sets are well perturbed. From the data utility results shown in Table II, there is no significant changes in overall correct predict rate. The interesting thing is that, after feature selection, the predictive power of SVM for FS is increased compared to the SVM

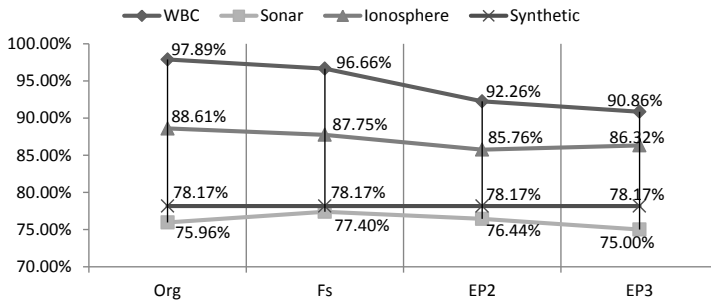

Fig. 6. SVM Rate Comparisons

results for original data (Org). This demonstrates feature selection's ability to rule out trivial values and noises. After removing those insignificant features, the data are "purified" and thus result in better mining performance.

D. Experiment 3

In comparison to experiment 2, we carried out the experiment 3 in a reversed sequence. We instead, distort original data using SVD first, and then select features on perturbed data. The parameter settings and configuration for sparsification and SVM are the same as in the Experiment 1 and Experiment 2 for consistency purpose.

The Figure 6 shows the SVM results for all the data sets in different experiments. As we can see, There is no significant differences in SVM predict rates for all experiments. From the feature selection's perspective, Experiment 3 has better results for both data sets. The sizes of selected feature space for both data sets have evident drops with only insignificant impacts on SVM results.

E. Summary

By comparing the empirical results, some important and interesting facts can be observed.

- The results in our experiments indicate that, for classification purpose, data owner publishing perturbed data before feature selection results in no significant difference in correct prediction rate than the other way around.
- Data distortion process should be done on selected feature space, since the discarded features by feature selection procedure have very little perturbation values and perturbing data with reduced feature space reduces computational expenses.
- Applying SSVD and performing sparsification process on small entries of decomposed matrices has potential to eliminate garbage information and improve mining qualities.
- From Feature Selection perspective, performing feature selection process after sparsification process by SSVD would result in better outcomes, i.e. more irrelevant features can be identified.

V. Conclusion And Future Plans

We conclude that performing feature selection before data perturbation is a better approach than the other way around for classification purpose, since there is no major distinguishable contrast in prediction outcomes and discarded features have little perturbation values. Furthermore, perturbing a data with reduced feature space is more cost-effective. On the other hand, the perturbed data published by data owner also have little effects on correct prediction rate, but could result in better feature selection results. Empirical tests are required for choosing the rank of SVD and setting proper threshold parameters. How many singular values to keep or how large a threshold should be set is different from applications to applications and, of course, is dependent on the nature of the data to be distorted. In the future, we would like to design and develop various algorithms or schemes in order to further exploring how we could use feature selection, in combined with matrix decomposition techniques, to serve privacy preserving data mining.

References

[1] Z. Yang, S. Zhong, R. N. Wright. *"Privacy-preserving classification of customer data without loss of accuracy"*, In Proceedings of the 5th SIAM International Conference on Data Mining, Newport Beach, CA, April 21-23, 2005

[2] J. Gao, J. Zhang. *"Sparsification strategies in latent semantic indexing"*, In Proceedings of the 2003 Text Mining Workshop, M.W. Berry and W.M. Pottenger, (ed.), pp. 93-103, San Francisco, CA, May 3, 2003

[3] S. Xu, J. Zhang, D. Han,J. Wang. *"Data distortion for privacy protection in a terrorist analysis system"*, In Proceedings of the 2005 IEEE International Conference on Intelligence and Security Informatics, pp. 459-464, Atlanta, GA, May 2005.

[4] J. Wang, W. Zhong, S. Xu, J. Zhang. *"Selective Data Distortion via Structural Partition and SSVD for Privacy Preservation"*, In Proceedings of the 2006 International Conference on Information & Knowledge Engineering, pp:114-120, CSREA Press, Las Vegas

[5] T. Marill, D. M. Green. *"On the effectiveness of receptors in recognition systems"*, IEEE Transations on Information Theory, 9:11-17, 1963.

[6] A. W. Whitney. *"A direct method of nonparametric measurement selection"*, IEEE Transactions on Computers, 20:1100-1103, 1971.

[7] Michael W. erry, Zlatko Drmač, Elizabeth and R. Jessup. *"Matrix,Vector space, and information retrieval"*, SIAM Rev 41:355-362, 1999.

[8] W. H. Wolberg and O. L. Mangasarian. *"Multisurface method of pattern separation for medical diagnosis applied to breast cytology"*, Proceedings of the National Academy of Sciences, U.S.A., Volume 87, December 1990, pp 9193-9196.

[9] R. P. Gorman and T. J. Sejnowski. *"Analysis of Hidden Units in a Layered Network Trained to Classify Sonar Targets"*, SIAM Rev 41:355-361

[10] P. Lin, J. Zhang, Ingrid St. Omer, H. Wang, and J. Wang. *"A comparative study on data perturbation with feature selection"*, Lecture Notes in Engineering and Computer Science: Proceedings of The International MultiConference of Engineers and Computer Scientists 2011, IMECS 2011, 16-18 March, 2011, Hong Kong, pp 454-459.

[11] 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.

[12] 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

[13] G. Golub and C. Van Loan. *"Matrix Computations"*, Johns Hopkins University Press (3rd edition), 1996
[14] V. G. Sigillito, S. P. Wing, L. V. Hutton, K. B. Baker. *"Classification of radar returns from the ionosphere using neural networks"*, Johns Hopkins APL Technical Digest, 10, 262-266, 1989