# Privacy Preserving Linear Discriminant Analysis from Perturbed Data*

Somnath Chakrabarti
Department of Information Systems
University of Maryland Baltimore County
Baltimore, MD
chakra1@umbc.edu

Zhiyuan Chen
Department of Information Systems
University of Maryland Baltimore County
Baltimore, MD
zhchen@umbc.edu

Aryya Gangopadhyay
Department of Information Systems
University of Maryland Baltimore County
Baltimore, MD
gangopad@umbc.edu

Shibnath Mukherjee†
Yahoo! Research and Development, India
University of Maryland Baltimore County
Baltimore, MD
shibnath@gmail.com

## ABSTRACT

The ubiquity of the internet not only makes it very convenient for individuals or organizations to share data for data mining or statistical analysis, but also greatly increases the chance of privacy breach. There exist many techniques such as random perturbation to protect the privacy of such data sets. However, perturbation often has negative impacts on the quality of data mining or statistical analysis conducted over the perturbed data. This paper studies the impact of random perturbation for a popular data mining and analysis method: linear discriminant analysis. The contributions are two fold. First, we discover that for large data sets, the impact of perturbation is quite limited (i.e., high quality results may be obtained directly from perturbed data) if the perturbation process satisfies certain conditions. Second, we discover that for small data sets, the negative impact of perturbation can be reduced by publishing additional statistics about the perturbation along with the perturbed data. We provide both theoretical derivations and experimental verifications of these results.

## 1. INTRODUCTION

The ubiquity of the internet makes it very convenient for individuals or organizations to share data for data mining or statistical analysis. For example, a research group can create an internal website and put data sets used in their research on the website. There are also websites such as Google Groups that allows a group of people to share data over the web.

Some of these data sets may contain privacy sensitive information such as medical conditions of a patient or financial status of a person. Using the web to share such data sets greatly increases the chance of privacy breach. For example, a researcher may accidentally give public access to the internal website, or the website may get hacked. Even if the data set is exposed for a short period of time, search engines such as Google may have already allowed many people to find that webpage and gain access to the data set. Of course, one can email the data set to his collaborators instead of using the web. The email server however may have a copy of the data and may get hacked as well. Therefore, sanitizing data sets before sharing them may reduce the chance of privacy breach. Further, when the sharing occurs across organizational boundaries (e.g., when a company hires some consultant to mine its customer data), the shared data sets often must be sanitized. Legislation such as the Health Insurance Portability and Accountability Act (HIPAA) also requires protection of privacy.

There exist many privacy protection techniques [1]. A popular method is random perturbation[3], which adds random noise to the data. However, such privacy protection methods typically distort the values of the original data, which may have negative impacts on the quality of data mining or statistical analysis conducted over the perturbed data.

One possible approach to reduce such negative impacts is to reconstruct the original data distribution from the perturbed data. For example, in [3, 2], the authors proposed an iterative method to reconstruct the distribution of the original data from the perturbed data. The reconstructed distribution can be used in subsequent data mining or statistical analysis process. The authors showed that this method

*This work was supported in part by the National Science Foundation under Grant Numbers IIS-0713345.
†This work was done when he was at UMBC.


*SAC'10* March 22-26, 2010, Sierre, Switzerland.

works well for decision tree classification. However, it is unclear whether this approach works for other mining tasks.

This paper studies the impact of random perturbation for a popular data mining and analysis method: linear discriminant analysis (LDA). LDA and the related Fisher's linear discriminant are methods used in statistics and machine learning to find the linear combination of features which best separate two or more classes of objects or events. The resulting combination may be used as a linear classifier or as a dimensionality reduction technique.

The contributions of this paper are two fold. First, we discover that for large data sets, the impact of perturbation is quite limited (i.e., high quality results may be obtained directly from perturbed data) if the perturbation satisfies certain conditions. Second, we discover that for small data sets, the negative impact of perturbation can be reduced by publishing additional statistics about the perturbation along with the perturbed data. Such statistics also do not compromise privacy. We provide both theoretical derivations and experimental verifications of these results.

The rest of the paper is organized as follows. Section 2 describes related work. Section 3 gives necessary background about LDA. Section 4 shows the theoretical derivations. Section 5 reports experimental results. Section 6 concludes the paper.

## 2. RELATED WORK

There has been a rich body of work on privacy preserving data mining. A good survey can be found at [1]. The commonly used techniques include random perturbation [3], random swapping [4], and generalization [14]. There has also been work on PPDM in distributed setting and survey articles can be found at [1]. This paper considers the centralized setting when the data owner (e.g., a company that collects the data) wants to outsource the data mining task to an outsider (e.g., a consultant).

It is well-known that random perturbation has negative impacts on the quality of mining and statistical analysis. A distribution reconstruction method was proposed in [3, 2] for classification. However, it is unclear whether the reconstructed global distribution will be sufficient for other mining tasks. There has also been work on using the statistical properties of the added noise in Naive Bayesian classification [11] and decision tree classification [10]. This paper also uses statistical properties of the added noise, but considers a different mining method: LDA and Fisher's LDA.

## 3. BACKGROUND OF LDA

LDA [12] and Fisher's LDA [6] are used to find a linear combination of features that best separate two classes. LDA assumes that the data points in each class follow normal distribution and have the same covariance, while Fisher's LDA does not have such assumptions. Hence LDA can be seen as a special case for Fisher's LDA. In this paper, we use Fisher's LDA in our analysis but our results also apply to LDA. To simplify notations, we use LDA and Fisher's LDA interchangeably in this paper.

Unlike Principal Component Analysis that finds the direction that maximizes the variances, LDA methods find the direction that best discriminates the two classes. For example, Figure 1 illustrates two classes (one represented as triangle and one represented as circle). The direction that

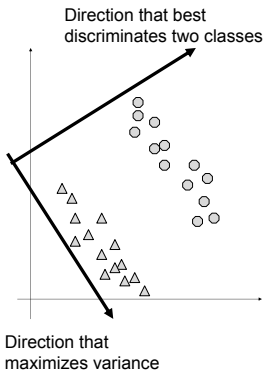


**Figure 1: LDA best discriminates two classes**

maximizes variance is different from the direction that best discriminates the two classes.

LDA methods can be also generalized to non linearly separable cases using Kernel methods. They can also be generalized to handle multiple classes [12]. They are widely used in business applications such as predicting bankruptcy, face recognition, and marketing research (e.g., to analyze results of surveys).

Let $D$ be a data set that contains $n$ records and $m$ attributes. Let $x_i$ be the $i$-th record in $D$ and it can be represented as a $m$-vector. Let $y_i$ be the class label of $x_i$, and it may have values 1, 2, ..., $k$ (i.e., there are $k$ classes). Let $C_y$ be the records in class $y$. Fisher LDA tries to find a $m$-vector $w$ that maximizes

$$J(w) = \frac{w^T S_B w}{w^T S_W w} \tag{1}$$

$J(w)$ is the ratio of between-class variance to the within-class variance. $S_B$ is the between class scatter matrix.

$$S_B = \sum_{y=1,\ldots,k} N_y(\mu_y - \mu)(\mu_y - \mu)^T \tag{2}$$

In Equation (2), $N_y$ is the number of records in class $y$, $\mu_y$ is the centroid of class $y$, and $\mu$ is the mean of all records. If we approximate each record $x_i$ with the centroid of the class that $x_i$ belongs to, $S_B$ can be also seen as the covariance matrix of this approximated data set, scaled by the number of records in the data set.

$S_W$ is the within class scatter matrix.

$$S_W = \sum_{y=1,\ldots,k} \sum_{x_i \in C_y} (x_i - \mu_y)(x_i - \mu_y)^T \tag{3}$$

Note that the covariance matrix of class $y$ is $\frac{1}{N_y}\sum_{x_i \in C_y}(x_i - \mu_y)(x_i - \mu_y)^T$. Thus $S_W$ is also the sum of covariance matrices of all classes, scaled by the number of records in each class. Let $S_B^{1/2}$ be the square root of matrix $S_B$, and $S_W^{-1}$ be the inverse of $S_B$. We need to compute a symmetric, positive definite matrix S

$$S = S_B^{1/2} S_W^{-1} S_B^{1/2} \tag{4}$$

The solution to Fisher's LDA can be obtained by first finding the largest Eigen value and Eigen vector for $S$, i.e., to find the largest value $\lambda$ and the corresponding $m$ by 1 vector $v$ such that

$$Sv = \lambda v \tag{5}$$

The final solution $w = S^{-1/2}v$. Once the $w$ is found, one can project all records to $w$, and find a value $b$ s.t. $w^T x + b > 0$ for records in class 1 and $w^T x + b < 0$ for records in class 2. In case there are overlap between class 1 and 2, one can choose $b$ to minimize the classification error.

# 4. THEORETICAL RESULTS

In this paper, we consider a centralized scenario where the data owner will send perturbed data to a data receiver, who will conduct data mining and then send back the data mining results.

Note that the solution of Fisher's LDA depends on the two scatter matrices $S_B$ and $S_W$. Let $S_B^*$ and $S_W^*$ be the between-class and within-class scatter matrices after noise addition, respectively. In Section 4.1, we will derive the relationship between $S_B^*$ and $S_B$, and the relationship between $S_W^*$ and $S_W$. In Section 4.2, we show that for large data sets, when the random perturbation process satisfies certain conditions, Fisher's LDA will have good results on the noisy data. In Section 4.3, we will show the results for smaller data sets.

## 4.1 Relationship between Noisy and Original Scatter Matrices

Let $\Delta_{xi}$ be the random noise added to record $x_i$. Let $\mu_y^*$ be the centroid of class $y$ after noise addition. Let $\mu^*$ be the global mean after noise addition. Let $\Delta_{\mu_y}$ be the mean of noise added to class $y$, and $\Delta_\mu$ be the mean of noise added to the whole data set. We have

$$\begin{aligned}\mu^* &= \frac{1}{N}\sum_{1\le i\le N}(xi + \Delta_{xi}) \\ &= \frac{1}{N}\sum_{1\le i\le N} xi + \frac{1}{N}\sum_{1\le i\le N}\Delta_{xi} \\ &= \mu + \Delta_\mu \end{aligned} \quad (6)$$

Similarly, we can prove that

$$\mu_y^* = \mu_y + \Delta_{\mu_y} \quad (7)$$

The between-class scatter matrix over perturbed data is:

$$\begin{aligned}S_B^* &= \sum_{y=1,\ldots,k} N_y(\mu_y^* - \mu^*)(\mu_y^* - \mu^*)^T \\ &= \sum_{y=1,\ldots,k} N_y(\mu_y + \Delta_{\mu_y} - \mu - \Delta_\mu)(\mu_y + \Delta_{\mu_y} - \mu - \Delta_\mu)^T \\ &= \sum_{y=1,\ldots,k} N_y(\mu_y - \mu + \Delta_{\mu_y} - \Delta_\mu)(\mu_y - \mu + \Delta_{\mu_y} - \Delta_\mu)^T \\ &= S_B + \Delta_B + S_{B\Delta} + S_{B\Delta}^T \end{aligned} \quad (8)$$

Here $\Delta_B = \sum_{y=1,\ldots,k} N_y(\Delta_{\mu_y} - \Delta_\mu)(\Delta_{\mu_y} - \Delta_\mu)^T$ and $S_{B\Delta} = \sum_{y=1,\ldots,k} N_y(\mu_y - \mu)(\Delta_{\mu_y} - \Delta_\mu)^T$. Since the perturbed data is $x_i + \Delta_{xi}$, we can see the perturbed data set as the combination of the original data set and a noise data set formed by added noise. Thus $\Delta_B$ can be seen as between class scatter matrix for the noise data set. $S_{B\Delta}$ can be seen as the covariance matrix between cluster centers of the original data set and cluster centers of the noisy data set, scaled by the number of records in the data set.

Similarly, we can compute the within-class scatter matrix after noise addition as:

$$\begin{aligned}S_W^* &= \sum_{y=1,\ldots,k}\sum_{x_i\in C_y}(x_i^* - \mu_y^*)(x_i^* - \mu_y^*)^T \\ &= \sum_{y=1,\ldots,k}\sum_{x_i\in C_y}(x_i + \Delta_{xi} - \mu_y - \Delta_{\mu_y}) \\ &\quad (x_i + \Delta_{xi} - \mu_y - \Delta_{\mu_y})^T \\ &= \sum_{y=1,\ldots,k}\sum_{x_i\in C_y}(x_i - \mu_y + \Delta_{xi} - \Delta_{\mu_y}) \\ &\quad (x_i - \mu_y + \Delta_{xi} - \Delta_{\mu_y})^T \\ &= S_W + \Delta_W + S_{W\Delta} + S_{W\Delta}^T \end{aligned} \quad (9)$$

Here $\Delta_W = \sum_{y=1,\ldots,k}\sum_{x_i\in C_y}(\Delta_{xi} - \Delta_{\mu_y})(\Delta_{xi} - \Delta_{\mu_y})^T$. $\Delta_W$ is the within class scatter matrix of the noise.

$S_{W\Delta} = \sum_{y=1,\ldots,k}\sum_{x_i\in C_y}(x_i - \mu_y)(\Delta_{xi} - \Delta_{\mu_y})^T$. The covariance matrix between data and noise in class $y$ is

$$\frac{1}{N_y}\sum_{x_i\in C_y}(x_i - \mu_y)(\Delta_{xi} - \Delta_{\mu_y})^T.$$

Thus $S_{W\Delta}$ is the sum of covariance matrix between data and noise in each class, scaled by the number of records in each class.

## 4.2 Results for Large Data Sets

As shown in [8, 9], noise addition techniques are vulnerable to attacks that use correlation of data. A solution was proposed in [13]. It first apply Principal Component Analysis over the data, and then adds noise in the PCA dimensions. On each PCA dimension, the variance of the added noise is also proportional to the range of data in that dimension. This method avoids the vulnerability of existing noise addition techniques because the data correlation has been removed.

In this paper, we assume that a PCA has been conducted before noise addition. We also assume that the noise has zero mean (which is the case in most existing work), and is added independently in each attribute. The noise is also independent of the data. We also assume that the variance of noise is proportional to the variance of data in each attribute.

Since the noise has zero mean, for large data sets, the mean of noise ($\Delta_\mu$) and the mean of noise in each class ($\Delta_{\mu_y}$) are also zero. Thus in Equation (8), $\Delta_B$ and $S_{B\Delta}$ are both zero. Thus we have

$$S_B^* = S_B \quad (10)$$

Similarly, since data and noise are independent in each class, $S_{W\Delta}$ is zero. The noise in different attributes is also independent of each other. Let $\sigma^2_{\Delta_{ij}}$ be the variance of noise in attribute $j$ and class $i$. $\Delta_W$ is a diagonal matrix where the $j$-th diagonal item equals $\sum_{i=1,\ldots,k} N_i\sigma^2_{\Delta_{ij}}$.

Further, since PCA has been applied first, the data attributes are independent of each other as well, thus $S_W$ is also a diagonal matrix where the $j$-th diagonal item in $S_W$ equals $\sum_{i=1,\ldots,k} N_i\sigma_{ij}^2$. Here $\sigma_{ij}^2$ is the variance of data in attribute $j$ and class $i$.

Since the variance of noise is proportional to the variance of data, we have $\sigma^2_{\Delta_{ij}} = r\sigma_{ij}^2$ where $r$ is a constant. Thus

$$S_W^* = S_W + S_{W\Delta} = S_W(1 + r) \quad (11)$$

Let $S^* = S_B^{*1/2} S_W^{*-1} S_B^{*1/2}$. Using Equation (10) and (11), we have

$$\begin{aligned} S^* &= S_B^{1/2}((1+r)S_W)^{-1}S_B^{1/2} \\ &= \frac{1}{1+r} S_B^{1/2} S_W^{-1} S_B^{1/2} \\ &= \frac{1}{1+r} S \end{aligned} \quad (12)$$

Since $S^*$ equals $S$ multiplied by a constant, we have proved that the solution $w^*$ to Fisher's LDA for the perturbed data equals the solution $w$ to the original data multiplied by a constant as well. The detail can be found in the Appendix. Since $w$ represents the direction to discriminate the two classes, multiplying it by a constant will not change the results.

## 4.3 Results for Small Data Sets

Let $Var(\Delta)$ be the variance of the noise. We have $Var(\Delta_\mu) = Var(\Delta)/N$. As $N$ (the number of records in the data set) decreases, the variance of noise mean $\Delta_\mu$ increases. Thus for a small data set, the mean of noise in the data set may not equal to zero due to small sample error. Similarly, the mean of noise in each class also may not be zero. The within class scatter matrix for data ($S_W$) and the scatter matrix for noise ($\Delta_W$) may not be diagonal matrices as well. Thus Equation (10) and (11) may not hold. Thus directly running LDA on perturbed data may not get good results.

**Sending additional statistics:** We let the data owner send the following statistical information about noise:

1. Mean of noise in each class ($\Delta_{\mu_y}$)
2. The within class scatter matrix for noise ($\Delta_W$)

The receiver can first compute $S_B^*$ and $S_W^*$ using perturbed data. He then computes $S_B$ and $S_W$ as follows

$$S_B = S_B^* - \Delta_B \quad (13)$$

$$S_W = S_W^* - \Delta_W \quad (14)$$

These two equations can be inferred from Equation (8) and (9) if we assume that $S_{B\Delta}$ and $S_{W\Delta}$ are zero. The global mean of noise $\Delta_\mu$ can be computed from $\Delta_{\mu_y}$. Then $\Delta_B$ can be computed using $\Delta_\mu$ and $\Delta_{\mu_y}$. $\Delta_W$ is provided by the owner.

In Section 4.1, we have shown that $S_{B\Delta}$ is the covariance matrix between cluster centers for data and cluster centers for noise. $S_{W\Delta}$ is also the sum of covariance between data and noise in each class. Since noise and data are independent, we assume that both matrices are zero. Another reason for doing this is that sending these two matrices may compromise privacy. For example, if noise and data are somehow highly correlated in a class (which will be reflected in $S_{W\Delta}$), one may infer the original data values from the perturbed values.

**Privacy aspects:** Let $m$ be the number of attributes and $N$ be number of records in the data set. There are $m^2$ entries in matrix $\Delta_W$. Let $\Delta_W^{ij}$ be the item at $i$-th row and $j$-column in $\Delta_W$. Attackers also know the mean of noise added to each class. Let $\Delta_{ij}$ be the noise added to attribute $j$ of record $x_i$ and $\Delta_{\mu_{ij}}$ be the $j$-th attribute of $\Delta_{\mu_i}$. Using the definition of mean of noise in each class, we have $k$ equations in the format of

$$\sum_{x_i \in C_y} \Delta_{ij} = \Delta_{\mu_{yj}}, \forall y, 1 \le y \le k$$

We also have $m^2$ equations for $\Delta_W$

$$\Delta_W^{ij} = \sum_{y=1,\ldots,k} \sum_{x_p \in C_y} (\Delta_{pi} - \Delta_{\mu_{yi}})(\Delta_{pj} - \Delta_{\mu_{yj}}) \forall i, j, 1 \le i, j \le m$$

Thus there are $m^2+k$ equations and $mN$ variables ($\Delta_{ij}, 1 \le i \le N, 1 \le j \le m$). If $N >> m$, there are many more variables than equations so one can not get the exact solution (i.e., the exact noise added). Without knowing the exact noise, one can not infer the original data value from the perturbed data.

We also assume that in the original data, each class contains at least a certain number of records. This prevents one from guessing the noise added to a record based on the mean and variance of the noise in that class. The data owner can select the appropriate threshold. In this paper we set the threshold to 100.

# 5. EXPERIMENTS

We have conducted experiments to verify the theoretical results presented in Section 4. We first describe the perturbation process. We then describe the setup of the experiments and the results.

## 5.1 Perturbation Process

We used the same perturbation process as the perturbation process proposed in [13]. We first apply a PCA to the original data. This removes data correlation and avoids attacks based on data correlation [8, 9]. We then select the first few PCA components because the remaining ones are close to zero. We then add noise independently to each attribute. The noise follows Laplace distribution, and the variance is proportional to the range of data in that attribute. That is, for attribute $j$, the noise $\delta$ follows distribution $f(\delta) = \frac{1}{2b} e^{\frac{-|\delta|}{(max(x_j) - min(x_j))b}}$ where $b$ is a given parameter (ranges from 0 to 1), and $\max(x_j)$ and $\min(x_j)$ are the maximum and minimum values of that attribute, respectively.

The perturbation method proposed in [13] provides a worst case privacy guarantee. The worse case guarantee prevents upward or downward privacy breach [5]. Here we give an example for upward breach. Suppose there is a privacy sensitive property $Q$ that is very rare in original data. Given the perturbed value of a certain record $P$, if one can infer that $P$ satisfies $Q$ with a much higher probability, an upward privacy breach occurs.

For example, suppose the original data contains the salary of employees. The probability of having an extremely high salary (say, over one million dollars) equals 0.001. Now, suppose we observe a perturbed salary of a person $P$, which is ten million dollars. One may infer that $P$'s salary is over one million with 0.99 probability. This is a 0.001 to 0.99 privacy breach.

The above perturbation method prevents such upward breach. More specifically, let $\rho_1$ be the probability of $Q$ in original data and $\rho_2$ be the probability of $Q$ given the perturbed value of $P$. [5] shows that there will be no $\rho_1$-to-$\rho_2$(upward) breach if

$$\frac{\gamma \rho_1}{1 + (\gamma - 1)\rho_1} \le \rho_2 \le 1$$

Here $\gamma$ is *amplification* which will be described shortly. The intuition is as follows. Let $x_1$ and $x_2$ be two values in

**Table 1: Properties of Data Sets**

| Data Set | Num. of records | Num of attributes |
|---|---|---|
| Adult | 30717 | 6 |
| Cancer | 569 | 30 |

the original data, and $y$ be a perturbed value. Let $P[x_i \rightarrow y]$ be the probability of mapping $x_i$ to y after perturbation. The amplification $\gamma$ is defined as the maximal possible ratio of the probability of mapping $x_1$ to $y$ to the probability of mapping $x_2$ to $y$. That is:

$$\gamma = \max_{x_1, x_2, y} \frac{P[x_1 \rightarrow y]}{P[x_2 \rightarrow y]}$$

We have proved in [13] that $\gamma$ only depends on the distribution of noise and the Laplace noise will give bounded $\gamma$. That is $\gamma = e^{1/b}$. Suppose $x_1$ satisfies $Q$ and $x_2$ does not satisfies $Q$. From the definition of $\gamma$, we have

$$P[x_2 \rightarrow y] \geq \frac{P[x_1 \rightarrow y]}{\gamma}$$

Thus a bounded $\gamma$ means the original values that do not satisfy $Q$ (i.e., $x_2$) may be also mapped to $y$ with a significant probability (which is the right hand side of the above formula). Thus one can not easily decide whether the original value satisfies $Q$ based on the perturbed value $y$ because those do not satisfy $Q$ may be mapped to $y$ as well. On the other hand, as shown in [13], normal or uniform noise do not give bounded $\gamma$ and thus provide no privacy guarantee.

Plugging in the maximum value of $\gamma$ as $e^{1/b}$ we know that there will be no privacy breach if

$$\frac{e^{1/b}\rho_1}{1 + (e^{1/b} - 1)\rho_1} \leq \rho_2 \leq 1$$

This means the worst case probability of privacy breach $\rho_2$ is at most $\frac{e^{1/b}\rho_1}{1+(e^{1/b}-1)\rho_1}$, irrespective of the $y$ value. Thus by making noise scale $b$ sufficiently large, we can guarantee small chance of privacy breach. For example, $b = 0.2$ in the perturbation process. No matter what value we observe (e.g., one billion dollars), the probability of predicting that $P$ earns over one million dollars will be less than 0.13.

## 5.2 Setup

We used the Adult data set and the Wisconsin Breast Cancer data set from UCI machine learning repository [7]. Both data sets contain 2 classes. For the Adult data set, we used the income attribute as class label (either exceeds 50K or below 50K). Since LDA only works for numerical attributes, we only used numerical attributes for both data sets. The properties of data sets are shown in Table 1. In the experiments, we selected 10 PCA components for Cancer data and 3 components for Adult data because these numbers gave the best accuracy results.

For each data set, we generated 10 pairs of testing and training data sets as 10-fold cross-validation. We assumed that the training data would be perturbed and sent to the outsider for LDA. For each pair of data sets, the training data was perturbed and a Fisher's LDA model was built from it. This model was then applied to the unperturbed test data and the classification accuracy was computed. Finally the accuracy of all testing sets was averaged. The testing data was not perturbed because when the data owner

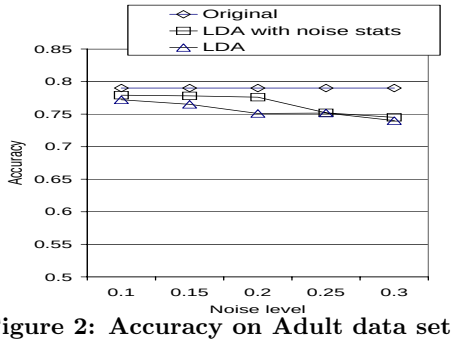


**Figure 2: Accuracy on Adult data set**

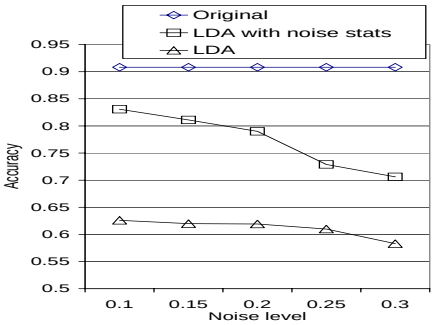


**Figure 3: Accuracy on Cancer data set**

applies a mining model on their own data set, he does not need to share his data. We used amplification to measure the worst case privacy (the smaller the value, the better the privacy protection). Since the perturbation process is also random, we repeat this process 20 times and report the average results.

We compared two methods, one running LDA directly on perturbed data, and the other using the statistical information about noise.

## 5.3 Experimental Results

Figure 2 and Figure 3 report the accuracy of LDA and LDA with statistics of noise on perturbed data. The accuracy over the original data is also shown as a baseline. We also varied the noise level $b$ from 0.1 to 0.3 for both data sets.

The results show that for Adult data set, the accuracy of both methods are quite close to that of the original data. Adult is a large data set (with 30717 records). Thus this verifies that for large data sets, directly running LDA on perturbed data gives good results.

The results also show that for Cancer data set, the accuracy of LDA with statistics on noise is significantly higher than that of LDA without the statistics. Cancer is a small data set (with just 569 records). This verifies that for small data sets, using statistical information on noise gives better results.

The accuracy of both methods decreases as noise level increases. This is expected because we ignore matrices $S_{B\Delta}$ and $S_{W\Delta}$ in Equation (8) and (9), and these two matrices have larger values for larger noise.

Figure 4 reports the amplification of both data sets as we vary the noise level. Since amplification only depends on noise level, the amplification for the same noise level is the same for both data sets. The results show that amplification decreases and the degree of privacy protection increases as noise level increases. For example, when noise level equals

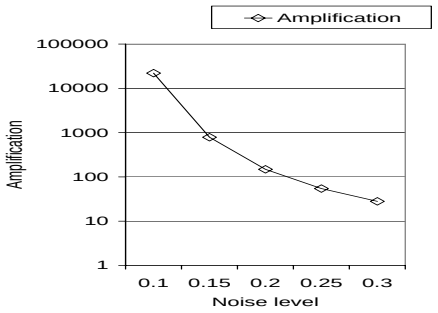


**Figure 4: Amplification for both data sets**

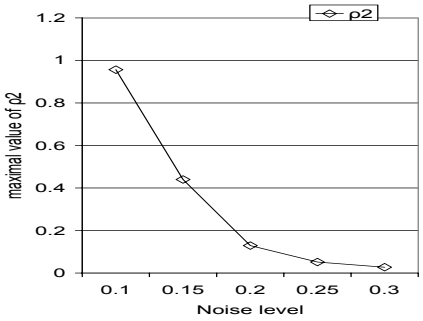


**Figure 5: Maximal value of $\rho_2$ if $\rho_1 = 0.001$**

0.2, the amplification is 148.

Suppose $\rho_1$, the probability of a privacy sensitive property in the original data, is 0.001. Figure 5 reports the maximum possible value of $\rho_2$ (the probability of the privacy sensitive property given perturbed data) for various noise levels. For example, for b = 0.3, the probability of this privacy sensitive property will not exceed 0.028 given the perturbed data. Typically most privacy-sensitive data properties (e.g., the data values themselves) in real life appear in original data with low probabilities. The results show that the proposed method does not increase the conditional probabilities of such properties too much thus effectively restricting worst case privacy breaches.

## 6. CONCLUSIONS

This paper studies the impact of random perturbation on LDA and Fisher's LDA methods. The results show that for large data sets, LDA will have good results using perturbed data. For small data sets, this paper proposes a method to use additional statistical information about the added noise to improve the results of LDA. Experimental results have verified our findings.

## 7. REFERENCES


[1] C. C. Aggarwal and P. S. Yu. *Privacy-Preserving Data Mining: Models and Algorithms*. Springer, 2008.

[2] D. Agrawal and C. C. Aggarwal. On the design and quantification of privacy preserving data mining algorithms. In *20th ACM SIGMOD SIGACT-SIGART Symposium on Principles of Database Systems*, pages 247–255, Santa Barbara, CA, 2001.

[3] R. Agrawal and R. Srikant. Privacy preserving data mining. In *2000 ACM SIGMOD Conference on Management of Data*, pages 439–450, Dallas, TX, May 2000.

[4] T. Dalenius and S. P. Reiss. Data-swapping: A technique for disclosure control. *Journal of Statistical Planning and Inference*, 6:73–85, 1982.

[5] A. Evfimevski, J. Gehrke, and R. Srikant. Limiting privacy breaches in privacy preserving data mining. In *22nd ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pages 211 – 222, San Diego, CA, June 2003.

[6] R. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936.

[7] S. Hettich, C. Blake, and C. Merz. UCI repository of machine learning databases, 1998.

[8] Z. Huang, W. Du, and B. Chen. Derivin private information from randomized data. In *SIGMOD 2005*, pages 37–48, Baltimore, MD, June 2005.

[9] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar. On the privacy preserving properties of random data perturbation techniques. In *ICDM*, pages 99–106, 2003.

[10] L. Liu, M. Kantarcioglu, and B. Thuraisingham. Privacy preserving decision tree mining from perturbed data. In *HICSS*, 2009.

[11] L. Liu, M. Kantarcioglu, and B. Thuraisingham. The applicability of the perturbation based privacy preserving data mining for real-world data. *Data Knowl. Eng.*, 65:2008, 5-21.

[12] G. J. McLachlan. *Discriminant Analysis and Statistical Pattern Recognition*. Wiley Interscience, 2004.

[13] S. Mukherjee, M. Banerjee, Z. Chen, and A. Gangopadhyay. A privacy preserving technique for distance-based classification with worst case privacy guarantees. *Data Knowl. Eng.*, 66(2):264–288, 2008.

[14] L. Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, 10:2002, 571-588.


## APPENDIX

**Proof for large data set:** Suppose $\lambda$ and $v$ are the largest Eigen value and the corresponding Eigen vector for $S$. We have $Sv = \lambda v$. Since $S^* = \frac{1}{1+r}S$, we have $S^*v = (\frac{1}{1+r}\lambda)v$. Thus $S$ and $S^*$ have the same Eigen vectors. The Eigen values in the perturbed data also equals the Eigen values in the original data multiplied by a constant.

The solution $w$ in original data equals $w = S^{-1/2}v$. The solution $w^*$ in the perturbed data equals $w^* = S^{*-1/2}v$. Since $S^* = \frac{1}{1+r}S$, we have

$$w = S^{-1/2}v = ((1+r)S^*)^{-1/2}v = \frac{1}{\sqrt[2]{1+r}}S^{*-1/2}v = \frac{1}{\sqrt[2]{1+r}}w^*.$$

Thus the original solution $w$ equals the solution computed from the perturbed data (using $S^*$) multiplied by a positive constant.