Correlated Differential Privacy: Feature Selection in Machine Learning

Tao Zhang, Tianqing Zhu*, Ping Xiong, Huan Huo, Zahir Tari, Wanlei Zhou

***Abstract*—Privacy preserving in machine learning is a crucial issue in industry informatics since data used for training in industries usually contain sensitive information. Existing differentially private machine learning algorithms have not considered the impact of data correlation, which may lead to more privacy leakage than expected in industrial applications. For example, data collected for traffic monitoring may contain some correlated records due to temporal correlation or user correlation. To fill this gap, we propose a correlation reduction scheme with differentially private feature selection considering the issue of privacy loss when data have correlation in machine learning tasks. The proposed scheme involves five steps with the goal of managing the extent of data correlation, preserving the privacy, and supporting accuracy in the prediction results. In this way, the impact of data correlation is relieved with the proposed feature selection scheme, and moreover the privacy issue of data correlation in learning is guaranteed. The proposed method can be widely used in machine learning algorithms which provide services in industrial areas. Experiments show that the proposed scheme can produce better prediction results with machine learning tasks and fewer mean square errors for data queries compared to existing schemes.**

***Index Terms*—Differential privacy, machine learning, data correlation, feature selection**

I. Introduction

CURRENTLY, machine learning becomes an indispensable tool to provide services for human beings in industrial applications, such as Internet of Things (IoT) [1] and smart cities [2]. One main data source used for machine learning in industry is from human's activities. For example, human's data are often collected via smart phones and these data are analyzed to provide some services in smart cities, such as traffic monitoring [3] and smart health [4]. Data collected from human usually contain some sensitive information, as the location information and health data in above examples. When these data are used for machine learning, individual privacy can be leaked [5].

As a popular technique for privacy preserving, differential privacy was first proposed by Dwork et al. [6]. Since then, differential privacy has attracted considerable attention because it provides a rigorous mathematical framework for preserving privacy. Recently, differential privacy is widely used to protect the privacy in industrial informatics, such as location privacy protection [3], [7], smart grids [8], [9] and multi-agent systems [10].

Much work has addressed the privacy issue in machine learning with differential privacy. Chaudhuri provided an output perturbation where the model was trained and then the noise was added to the output [11] and objective perturbation mechanism where a carefully designed linear perturbation item was added to the original loss function [12]. [13] derived differentially private stochastic gradient descent mechanisms and tested them empirically in logistic regression. [14] proposed a differentially private deep learning algorithms which was based on a differentially private version of stochastic gradient descent. [15] studied the differentially private publishing model. However, previous works have not considered the data correlation when designing differentially private machine learning algorithms.

In the definition of differential privacy, data in a dataset are assumed to be independent. This is a somewhat faulty assumption since data in industrial applications are always correlated beginning from when the data is first generated, such as temporal datasets in monitoring systems. Intuitively, when some of the records in a dataset are correlated, deleting one record may have a great impact on the other records, which could reveal more information to an adversary than expected. Kifer and Machanavajjhala's study on data correlation [16] confirms this observation, and the finding has launched a new stream of research on how to preserve privacy in correlated datasets. [17], [18] introduced correlation parameters to describe data correlation. Correlation models were proposed to model data correlation, such as the Gaussian correlation model in [19], [20] and Markov chain models in [21]. Also, [22] designed a second privacy framework, called Pufferfish, which is flexible and can provide a privacy guarantee for various data sharing needs.

Correlated data used for industrial applications can also disclose more privacy information in machine learning algorithms when applying differential privacy. Previous methods do not always guarantee good performance because data correlation is not always easy to capture or describe accurately in the real world. Unlike previous studies, the proposed scheme correlation reduction based on feature selection (CR-FS) reduces data correlation and can be applied to both data analysis and data publishing, which provides a widely used applications in industries. Feature selection is a key method in machine learning for choosing the features that are crucial to predicting a result [23]. It is used to reduce overfitting, but can also be used to reduce data correlation across an entire dataset.

Tao Zhang, Tianqing Zhu, Huan Huo, Wanlei Zhou are with the School of Computer Science, University of Technology, Sydney, Australia. Email: {Tao.Zhang-3@student.uts.edu.au, Tianqing.Zhu@uts.edu.au, Huan.Huo@uts.edu.au, Wanlei.Zhou@uts.edu.au}

P. Xiong is with the School of Information and Safety Engineering, Zhongnan University of Economics and Law, Wuhan, China. Email: {pingxiong@znufe.edu.cn}

Zahir Tari is with the School of Computer Science and Software Engineering, RMIT University, Melbourne, Australia. Email: {zahir.tari@rmit.edu.au}

Overall, the contributions of this paper can be summarized as follows:

- 1) We proposed a differentially private feature selection based on feature importance. The proposed method can select features privately, while retaining a desirable data utility.
- 2) We propose a correlation reduction scheme based on feature selection to reduce data correlation in correlated datasets. This helps to reduce the correlated sensitivity when implementing differentially private machine learning algorithms, and thus improves data utility.
- 3) Experiments validate the effectiveness of our proposed feature selection scheme. The results show improved data utility for both data analysis and data publishing.

II. Preliminaries

A. *Differential privacy*

Differential privacy is a rigorous privacy model [24]. In brief, given two datasets D and $D^{'}$ that contains a set of records, these are referred as neighboring datasets when they differ in one record. A query Q is a function that maps the record $r \in D$ into outputs $Q(D) \in \mathcal{R}$, where $\mathcal{R}$ is the whole set of outputs.

Definition 1: (ϵ-Differential privacy) A randomized algorithm M satisfies *ϵ-differential privacy* if for any pair of datasets, say D and $D^{'}$, and for any possible outcome $Q(D) \in R$, we have

$$Pr[\mathcal{M}(D) \in R] \leq exp(\epsilon) \cdot Pr[\mathcal{M}(D^{'}) \in R] \qquad (1)$$

where ϵ refers to the privacy budget that controls the privacy level of the mechanism $\mathcal{M}$. The lower ϵ represents the higher privacy level.

Definition 2: (Sensitivity) For a query $Q : D \rightarrow \mathcal{R}$, and neighboring datasets, the sensitivity of Q is defined as

$$\Delta f = \max_{D,D^{'}} ||Q(D) - Q(D^{'})||_1 \qquad (2)$$

Sensitivity describes the maximal difference between neighboring datasets, which is only related to the type of query Q.

Definition 3: (Laplace mechanism) For any query Q: $D \rightarrow \mathcal{R}$ over the database D, the following mechanism provides ϵ-differential privacy if

$$\mathcal{M}(D) = Q(D) + Laplace(\Delta/\epsilon) \qquad (3)$$

The Laplace noise is denoted as $Laplace(\cdot)$ and is drawn from a Laplace distribution with the probability density function $p(x|\lambda) = \frac{1}{2\lambda}e^{-|x|/\lambda}$, where λ relate to the sensitivity and the privacy budget.

Theorem 1: Sequential composition: Suppose that a set of privacy mechanisms $\mathcal{M}$=$\{\mathcal{M}_1, ..., \mathcal{M}_m\}$, gives ϵ_i differential privacy ($i = 1, 2..., m$), and these mechanisms are sequentially performed on a dataset. $\mathcal{M}$ will provides $(\sum_i \epsilon_i)$-differential privacy for this dataset.

B. *Feature selection*

Feature selection is a method for selecting the attributes in a dataset (such as columns in tabular data) that are most relevant to the prediction [25]. In other words, feature selection largely acts as a filter that sifts out features that are less useful to solving a problem. With feature selection, both the efficiency and the accuracy of the predicted results can be improved.

In this paper, we adopt feature importance to select features. Feature importance is a method of ranking features based on random forests. Feature importance is measured according to the mean decrease in impurity, which is defined as the total decrease in node impurity averaged over the forest. This score can be computed automatically for each feature after training and scaling the results so that the sum of importance for all features is equal to 1. One strength of the random forest is that it is easy to measure which features are relatively more important to the results. With this method, we are able to select the most important features in the dataset.

III. Example of the traffic monitoring

In this section, we present the issue of correlated data in differential privacy with a detailed industrial example of traffic monitoring and show how correlated data can degrade the level of privacy in industry applications.

The traffic monitoring is one of most used technologies in smart cities. User's location information in a region are collected by a trusted server and the aggregate information of the dataset (i.e., the counts of users at each location) is continuously released to the public. Some users in the region may have a form of social relationship – perhaps family members. In this case, some users may have the same location information during some time and hence the records of users' information can be correlated in the dataset.

As shown in Table I, the user's locations are recorded at different time points. It is assumed that users only appear in one location at each time point, and it is observed that $user_1$ and $user_2$ take the same route from time point $t = 1$ to $t = 4$ (they may have social relationships). In this case, if one were to change the location of $user_1$, the location of $user_2$ would also change. In this way, the records for $user_1$ and the records for $user_2$ are correlated.

TABLE I: Users' locations at different times

t user	1	2	3	4
u_1	loc_2	loc_2	loc_3	loc_4
u_2	loc_2	loc_2	loc_3	loc_4
u_3	loc_1	loc_4	loc_5	loc_2
u_4	loc_4	loc_5	loc_2	loc_5

Table II shows that the some counts at different locations are always 2. In terms of the Laplace mechanism, adding the amount of $Lap(1/\epsilon)$ noise to perturb each count in Table 2 can achieve ϵ-DP at each time point. However, the expected privacy guarantee may breach with correlated records in the dataset. With background information of who has the relationship in a certain region, an attack can infer the location information of $user_1$ and $user_2$ at different time points. Consequently,

TABLE II: The sum counts of users' locations

t loc	1	2	3	4
loc_1	1	0	0	0
loc_2	2	2	1	1
loc_3	0	0	2	0
loc_4	1	1	0	2
loc_5	0	1	1	1

after releasing private count of user's locations, the location information of $user_1$ and $user_2$ may not be ϵ- differentially private as expected. Instead, it is 2ϵ-differentially private since changing one user's location will change the count 2.

In summary, this example shows that correlated data in a dataset will disclose more information than expected when these data are used for machine learning algorithms in industrial applications. Essentially, adding more noise to a correlated dataset is a way to guarantee differential privacy. Such a case reveals the level of challenge in industries when dealing with correlated data in situations where differential privacy must be satisfied, but high-quality query results must be maintained.

IV. The extent of data correlation

A. Correlated degree

Inspired by [17], we have incorporated the notion of correlated degree $\theta_{ij} \in [-1, 1]$ to denote the extent of correlation between record i and record j. When $|\theta_{ij}| > 0$, record i and record j have a positive correlation and vice versa. When $|\theta_{ij}| = 1$, record i and record j are fully correlated and When $\theta_{ij} = 0$, there is no relationship. When there are a number of l records in a dataset, it is possible to list the relationship for all records and form a correlated degree matrix Λ.

$$\Lambda = \begin{pmatrix} \theta_{11} & \theta_{12} & \cdots & \theta_{1l} \\ \theta_{21} & \theta_{22} & \cdots & \theta_{21} \\ \vdots & \vdots & \ddots & \vdots \\ \theta_{l1} & \theta_{l2} & \cdots & \theta_{ll} \end{pmatrix} \tag{5}$$

A threshold θ_0 is defined so as to select strongly correlated records. For a given θ_0, the value of the correlated degree is

$$\theta_{ij} = \begin{cases} \theta_{ij}, & \theta_{ij} \geq \theta_0, \\ 0, & \theta_{ij} < \theta_0, \end{cases} \tag{6}$$

A correlated degree matrix can describe the correlations of the whole dataset and, once analyzed, the curator will hold all knowledge of the data correlations. Data privacy can still be protected, even when the adversary is privy to the entire correlated degree matrix, if enough noise is added to mask the highest impact of deleting one record using correlated differential privacy.

B. Correlated sensitivity

Global sensitivity can only measure the maximal number of correlated records but does not consider the extent of the data correlation. Hence, the notion of correlated sensitivity is introduced to measure the extent of the impact on other records from changing one record. As mentioned earlier, global sensitivity adds extra noise by simply multiplying the maximal number of correlated records. Whereas, correlated sensitivity is able to model the correlations in a more exact way.

Definition 4: (Correlated sensitivity) For a query Q, correlated sensitivity is based on the correlated degree and the number of correlated records, which is defined as

$$\Delta CS_q = \max_{i \in q} \sum_{j=0}^{l} |\theta_{ij}| \{\|(Q(D^j) - Q(D^{-j})\|_1\} \tag{7}$$

where q is the set of records in a dataset, and θ_{ij} is the correlated degree between record i and record j. D_j and D_{-j} are neighboring datasets that differ by record j. Correlated sensitivity lists all the sensitivity of records with the query Q. With correlated sensitivity, the maximal effect on all records of a dataset can be measured when one record is deleted. For any query Q, the perturbed answer is calibrated with the equation,

$$\hat{Q}(D) = Q(D) + Laplace(\frac{\Delta CS_q}{\epsilon}) \tag{8}$$

For any query Q, the correlated sensitivity is smaller than the global sensitivity. The global sensitivity is denoted as $\Delta GS_q = \max_{i \in q} \sum_{j=0}^{k} \{k\|(Q(D^j) - Q(D^{-j})\|_1\}$, where k denotes the number of correlated records. Since we use the correlated degree $\theta_{ij} \in [-1, 1]$ to describe the extent of data correlation, the correlated sensitivity is no larger than the global sensitivity.

We note that the correlated degree θ_{ij} is related to every feature in record i and record j. When deleting features in the dataset, the extent of correlation between record i and record j will also be changed. Thus, after describing the extent of data correlation in a dataset, we use feature selection to reduce data correlation.

V. Correlation reduction based on feature selection

A. Overview of the method

In our method, we select features in terms of three principles: 1) the accuracy of training results; 2) the privacy of feature selection; 3) the reduction of the data correlation. As Fig. 1 shows, the proposed scheme CR-FS involves five steps: 1) removing collinear features; 2) removing unimportant features; 3) choosing features with differential privacy; 4) obtaining the Best feature set $\mathcal{B}$; and 5) adjusting the features that can reduce data correlation within the dataset. Each of these methods is described in detail in the following sections.

B. The proposed CR-FS scheme

Following traditional feature selection, we propose the algorithm I that selects features with differential privacy. For a given dataset, feature selection is a crucial step before executing a machine learning algorithm, especially with high-dimensional datasets. Additionally, retaining more features typically leads to a higher degree of data correlation, which, with differential privacy, negatively impacts the privacy level. Hence, our goal is to select a subset of features with relatively lower levels of data correlation while maintaining good utility for data publishing and analysis.

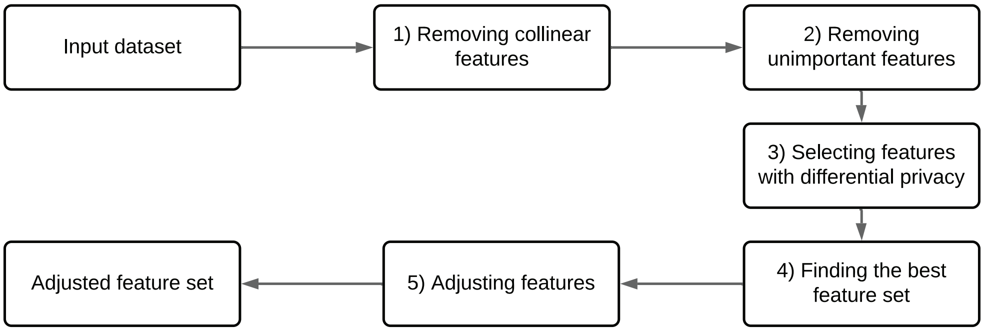

Input dataset
1) Removing collinear features
2) Removing unimportant features
3) Selecting features with differential privacy
4) Finding the best feature set
5) Adjusting features
Adjusted feature set

Fig. 1: The process of feature selection

Algorithm 1 Differentially private feature selection scheme

```
Input: Dataset, $T_{cf}, T_{fi}, T_{mv}$, $\epsilon_1$;
Output: Best feature set $\mathcal{B}$, Adjusted feature set $\mathcal{A}$;
Calculate feature collinearity $\rho_{f_m,f_n} = \frac{E[(f_m-\mu_{f_m})(f_n-\mu_{f_n})]}{\sigma_{f_m}\sigma_{f_n}}$; /* Step 1 */
if $\rho_{f_m,f_n} \leq T_{cf}$ then
    Remove $f_m$ or $f_n$;
end if
Remove unimportant features with $T_{fi}$; /* Step 2 */
Remove missing values with $T_{mv}$
Calculate the $fim_n$ of features by Random forest; /* Step 3 */
Calculate the sensitivity $\Delta fim$ according to Equation (11);
for $fim_n$; n=1,2,...,N: do
    Add Laplace noise $\hat{fim}_n = fim_n + Lap(\frac{\Delta fim_q}{\epsilon_1})$;
end for
Do the normalization $fim_n = \hat{fim}_n / \sum_{n=1}^{N} \hat{fim}_n$;
for i=1,2,...,n: do /* Step 4 */
    Delete features one by one according to the sequence of feature importance and calculate the prediction;
end for
Find the Best feature set : $\mathcal{B} = \{f_1, f_2, ...f_k\}$ and Adjusted feature set: $\mathcal{A} = \{f_{k+1}, ..., f_n\}$;
Add or delete features from Adjust feature set $\mathcal{A}$ according to algorithm 2;
```

1) Removing collinear features: The first step is to filter out the collinear features that can decrease generalization performance on the test set due to less model interpretability and high variance. Usually, the extent of collinearity between features is calculated by the absolute magnitude of the Pearson's correlation coefficient. The calculation of Pearson's correlation coefficient is

$$\rho_{f_m,f_n} = \frac{E[(f_m - \mu_{f_m})(f_n - \mu_{f_n})]}{\sigma_{f_m}\sigma_{f_n}} \tag{9}$$

Where f_m and f_n are two random features in the dataset; μ_{f_m} and μ_{f_n} are the mean of f_m and f_n; σ_{f_m} and σ_{f_m} are the standard deviation of feature f_m and f_n. In our scheme, we set a threshold of $T_{cf} \in [0, 1]$ to identify collinear features and remove the features with a collinearity of greater than T_{cf}.

2) Removing unimportant features: The second step is to remove unimportant features, including 1) features of zero importance and features of low importance; 2) features with a high percentage of missing values; and 3) features with a single value. Zero and low importance features can be identified using the feature importance threshold, denoted as $T_{fi} \in [0, 1]$. Features with an importance value of lower than T_{fi} will be removed. The threshold for missing values is defined as $T_{mv} \in [0, 1]$, and features with a percentage of missing values greater than T_{mv} will be removed.

3) Choosing features with differential privacy: We adopt feature importance fim in Random forest to calculate the feature weight for each feature. Neighboring data is obtained when record r_i is deleted, the feature importance can be calculated by Random forest and the feature importance $fim_1^i, fim_2^i, ..., fim_N^i$ are sorted in an increasing order. Based on this, we introduced the notion of record sensitivity of feature importance.

Definition 5: (Record sensitivity of feature importance) For a query Q, the record sensitivity of feature importance of r_i can be defined as,

$$\Delta fim_i = ||fim_N^i - fim_1^i||_1 \tag{10}$$

Definition 6: (Sensitivity of feature importance) For a query Q, the sensitivity of feature importance is determined by the maximal record sensitivity of feature importance,

$$\Delta fim_q = \max_{i \in q}(\Delta fim_i) \leq 1 \tag{11}$$

where q is a set of records related to a query Q. It is easy to know the sensitivity of feature importance is $\Delta fim_q \leq 1$, since the range of feature importance is from 0 to 1. We apply Laplace mechanism to add noise to the feature importance. The perturbed feature importance can be denoted as,

$$\hat{fim}_n = fim_n + Lap(\frac{\Delta fim_q}{\epsilon}) \tag{12}$$

Since the sum of the feature importance $\sum_{n=1}^{N} fim_n = 1$, we normalize the perturbed feature importance as follow,

$$fim_n = \hat{fim}_n / \sum_{n=1}^{N} \hat{fim}_n \tag{13}$$

The new sequence of feature importance can be denoted as $fim_1 < fim_2 < ... < fim_n$.

4) Finding the best feature set: The third step is to find the best feature set. The Best feature set $\mathcal{B}$ contains the features that will produce the best prediction results by the machine learning algorithm. In our method, the less important features are deleted one by one in the order of feature importance until the best chance of accurate predictions is achieved. Practically, finding Best feature set with this method demands far less computational overhead than other methods. The features that have not been selected for Best feature set are stored as the Adjusted feature set. These features can be used later for a tradeoff between utility and privacy. The Best feature set $\mathcal{B}$ can be denoted as $\mathcal{B} = \{f_1, f_2, ..., f_k\}$ and the Adjusted feature set $\mathcal{A}$ can be denoted as $\{f_{k+1}, f_{k+2}, ..., f_N\}$.

5) Adjusting feature scheme: The final step is to adjust some features based on the Best feature set $\mathcal{B}$ in order to reduce data correlation over the whole dataset, as a way to balance the tradeoff between utility and correlated sensitivity. Basically, the correlated sensitivity of a dataset is irrelevant to the number of features. This means that more features of a dataset may have a lower correlated sensitivity and less features may have a higher correlated sensitivity. Best feature set $\mathcal{B}$ can achieve a good data utility without privacy guarantee, yet it may have a higher correlated sensitivity and a high correlated sensitivity has a huge impact on utility for data publishing and data analysis. In other words, if the goal is to generate a differentially private dataset with good data utility, the process of feature selection should also consider correlated sensitivity.

Algorithm 2 Adjusted feature selection scheme

```
Input: Best feature set $\mathcal{B}$, Adjusted feature set $\mathcal{A}$, $\epsilon_2, \theta_0$;
Output: Adjusted feature set $\mathcal{A}$;
for $f_i \subseteq \{f_{k+1}, ..., f_N\}$: do
    Add features to the Best feature set $\mathcal{B}$ from the Adjusted feature set $\mathcal{A}$;
    Calculate the correlated sensitivity of new datasets $\Delta CS_q = \max_{i \in q} \sum_{j=0}^{l} |\theta_{ij}| \{\|(Q(D^j) - Q(D^{-j})\|_1$;
    Add Laplace noise $Lap = \frac{\Delta CS_q}{\epsilon_2}$;
    Train the dataset and get the predicted result;
end for
Obtain the Adjusted feature set $\mathcal{A}_1$ that has the best performance;
for $f_i \subseteq \{f_1, ..., f_k\}$: do
    Delete features from the Best feature set $\mathcal{B}$ one by one;
    Calculate the correlated sensitivity of new datasets $\Delta CS_q = \max_{i \in q} \sum_{j=0}^{l} |\theta_{ij}| \{\|(Q(D^j) - Q(D^{-j})\|_1$;
    Add Laplace noise $Lap = \frac{\Delta CS_q}{\epsilon_2}$;
    Train the dataset and get the predicted result;
end for
Obtain the Adjusted feature set $\mathcal{A}_2$ that has the best prediction;
if $s(\mathcal{A}_1) \geq s(\mathcal{A}_2)$ then
    $\mathcal{A}_1$ is the Adjusted feature set $\mathcal{A}$;
else
    $\mathcal{A}_2$ is the Adjusted feature set $\mathcal{A}$;
end if
```

Algorithm 2 shows the adjusted feature selection scheme, which includes backward and forward feature selection methods. The forward feature selection adds features one by one from the Adjusted feature set $\mathcal{A}$ to Best feature set $\mathcal{B}$. The correlated sensitivity is calculated according to Equation (7), and then Laplace noise is added according to Equation (8). Training with these added features can obtain the feature set $\mathcal{A}_1$, which provides optimal performance. However, sometimes adding a large number of features only slightly increases performance, particularly with high dimension datasets, while too many features can lead to a less interpretive model. Hence, when a set of added features appears to be more or less equally good, then it makes sense to choose the simplest feature set. We set a threshold T to evaluate the difference of training results. If the difference of training results is smaller than T, we select the simplest feature set that has the smallest number of predictors.

In backward feature selection, features in set are deleted one by one according to their feature importance. By comparing the training results with different deleted features, feature set $\mathcal{A}_2$ is generated, which has the best performance. Similar to forward feature selection, when a set of deleted features appears to be more or less equally good, it makes sense to choose the simplest feature set. We also use the threshold T to select the simplest feature set. Ultimately, the Adjusted feature set $\mathcal{A}$ is determined by comparing the training result $s(\mathcal{A}_1)$ and $s(\mathcal{A}_2)$.

C. Discussion

Best feature subset $\mathcal{B}$ and Adjusted feature set $\mathcal{A}$, represent the balance between utility and correlated sensitivity. Adding the adjusted features is likely to degrade data utility somewhat, but these extra features serve to reduce the correlated sensitivity of the dataset, which offsets the reduction in utility. The overall result is a feature selection scheme that strikes a balance that leads to less data correlation while maintaining good data utility for data analysis and data publishing.

Our proposed scheme has three advantages. First, feature importance is a computationally-efficient method for generating the best feature set compared to some of the other existing methods. Feature importance is the variable that provides the guide to select which features are best to add or delete. Second, with differential privacy, we can choose features privately. Third, with the consideration of data correlation, we can select features that has less data correlation in the whole dataset and thus reduce the correlated sensitivity and improve the data utility of the dataset.

VI. Theoretical analysis

A. Privacy analysis

Theorem 2: The proposed CR-FS scheme satisfies ϵ-differential privacy.

To prove that the proposed CR-FS scheme is satisfied with differential privacy, we first analyze which steps consume privacy budget in CR-FS scheme. According to Algorithm 1 and Algorithm 2, we access the dataset in two places: 1) the process of feature selection and, 2) the process of data training. To protect the data privacy, we add differential privacy noise in these two places.

We split the total privacy budget ϵ into two parts ϵ_1 and ϵ_2 and allocate ϵ_1 and ϵ_2 in the process of feature selection and the process of data training, respectively. First, we analyze the privacy budget ϵ_1 in the process of feature selection.

Lemma 1: The process of feature selection satisfies ϵ_1-differential privacy.

We know that D and $D^{'}$ are any two datasets that differ in one feature, and $f_1(\cdot)$ is the query for feature selection. $p_x(z)$ and $p_y(z)$ denote the probability density function as,

$$\mathcal{M}_1(x, f_1(\cdot), \epsilon_1) = f_1(x) + Lap(\frac{\Delta fim_q}{\epsilon_1}) \tag{14}$$

Let x, y be two neighboring datasets. We compare two random points $z \in \mathbb{R}$ and the ratio of two probability density can be presented as

$$\begin{aligned} \frac{p_x(z)}{p_y(z)} &= \prod_{i=1}^{N} \left(\frac{\exp\left(-\frac{\varepsilon_1 |f_1(x)i - z_i|}{\Delta fim_q}\right)}{\exp\left(-\frac{\varepsilon_1 |(f_1(y)i - z_i|}{\Delta fim_q}\right)} \right) \qquad (15) \\ &= \prod_{i=1}^{N} \exp\left(\frac{\varepsilon_1 \left(|f_1(y)_i - z_i| - |f_1(x)_i - z_i|\right)}{\Delta fim_q} \right) \\ &\leq \prod_{i=1}^{N} \exp\left(\frac{\varepsilon_1 |f_1(x)_i - f_1(y)_i\|}{\Delta fim_q} \right) \\ &= \exp\left(\frac{\varepsilon_1 \cdot \|f_1(x) - f_1(y)\|_1}{\Delta fim_q} \right) \\ &\leq \exp(\varepsilon_1) \end{aligned}$$

where the first inequality is from triangle inequality and the second inequality is from Equation (11). The sensitivity of feature selection is according to the maximal difference of feature importance. Therefore, the process of feature selection satisfies ϵ_1-differential privacy. Second, we analyze the privacy budget ϵ_2 in the process of data training.

Lemma 2: The process of data training satisfies ϵ_2-differential privacy.
We know that D and $D^{'}$ are any two datasets that differ in one record. $f_2(\cdot)$ is the query for training results.The differential privacy noise is added to the weights in training algorithms, such as Linear Regression (LR) and Support Vector Machine (SVM). $f_2(\cdot)$ is the query for the training results. We use $v_x(z)$ and $v_y(z)$ to denote the probability density function as,

$$\mathcal{M}_2(x, f_2(\cdot), \epsilon_2) = f_2(x) + Lap(\frac{\Delta CS_q}{\epsilon_2}) \qquad (16)$$

The ratio of two probability density can be presented as

$$\begin{aligned} \frac{v_x(z)}{v_y(z)} &= \prod_{i=1}^{N} \left(\frac{\exp\left(-\frac{\varepsilon_2 |f_2(x)i - z_i|}{\Delta CS_q}\right)}{\exp\left(-\frac{\varepsilon_2 |(f_2(y)i - z_i|}{\Delta CS_q}\right)} \right) \qquad (17) \\ &= \exp\left(\frac{\varepsilon_2 \cdot \|f_2(x) - f_2(y)\|_1}{\Delta CS_q} \right) \\ &\leq \exp(\varepsilon_2) \end{aligned}$$

The $\Delta CS_q = \max\limits_{i \in q} \sum_{j=0}^{l} |\theta_{ij}| \{ \|(Q(D^j) - Q(D^{-j})\|_1 \}$, hence the data training satisfies ϵ_2-differential privacy.

In the CR-FS scheme, we add privacy budget ϵ_1 and privacy budget ϵ_2 sequentially. Combined with **Lemma1**, **Lemma2** and **Theorem 1**, we can prove that the proposed CR-FS scheme satisfies $\{\epsilon_1 + \epsilon_2\}$-differential privacy.

VII. Experiments

Our evaluation experiments involve four real-world datasets in terms of both data analysis and data publishing tasks [26]. Utility for data analysis is tested with two machine learning algorithms: LR and linear SVM. Utility for data publishing is tested on count and mean queries.

A. Experimental setup

1) Dataset: The experiments involve four datasets, which have different extent of data correlation and different number of features.

- Adult Dataset [27]: Adult Dataset is from the UCI Machine Learning repository. After data preprocessing, we extract 3000 records with 12 features.
- Breast cancer Dataset [28]: This dataset can be found on UCI Machine Learning Repository. After data preprocessing resulted in 569 records with 20 features.
- Titanic Dataset [29]: This dataset comes from a Kaggle competition where the goal was to analyze which sorts of people were likely to survive the sinking of the Titanic. After data preprocessing, we extract 891 records with 9 features.
- Porto Seguro Dataset [30]: Porto Seguro is a well-known auto and homeowner insurance company. After preprocessing, we extract 1770 records with 37 features.

2) Comparison: For better comparisons, four schemes are considered in the experiments.

- A non-private scheme, where the dataset has no privacy protection.
- The group scheme, where noise is added by multiplying the number of correlated records, as proposed by Chen et al. in [31].
- The Zhu scheme, where noise is added according to the correlated sensitivity [17].
- The proposed scheme, where noise is added according to the CR-FS scheme defined in this paper.

3) Parameters: For correlation knowledge between records, no dataset suggests pre-defined knowledge of any correlated data. We use Pearson correlation coefficient to construct the correlated degree matrix, where a correlation exists for record i and record j if $\theta_{ij} \geq \theta_0$. θ_0 is set to 0.9 for Adult Dataset, Breast cancer Dataset and Breast cancer Dataset and θ_0 in Porto Seguro Dataset is set to 0.7. For correlation knowledge between features, the Pearson correlation coefficient threshold T_{fi} is set to 0.9. The missing value threshold T_{mv} is set to 0.2. The threshold of feature importance T_{fi} is set to 0.9.

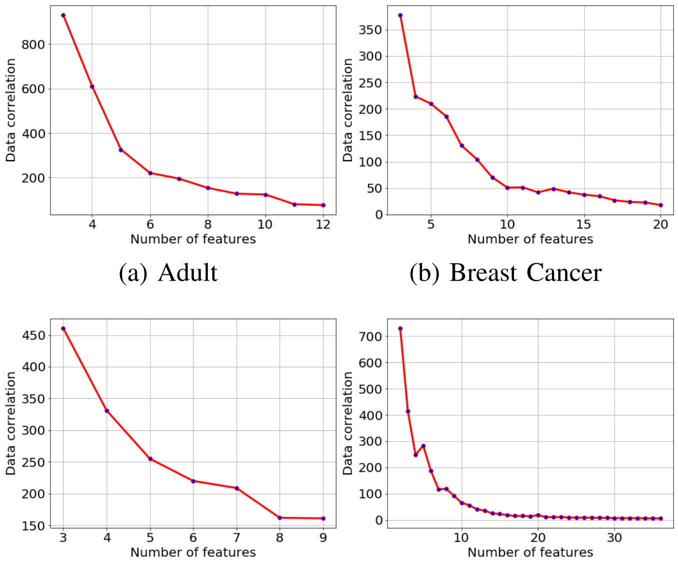
TABLE III: Number of features in different stages

	Original dataset	After data preparation	Best feature set $\mathcal{B}$	Adjusted feature set $\mathcal{A}$
Adult	15	12	8	12
Breast cancer	32	20	10	17
Titanic	12	9	7	9
Porto seguro	59	37	14	28

B. Experiments for data analysis

One aim of our proposed scheme is to improve utility for data analysis, which we evaluate according to the accuracy of the predicted results. For this set of experiments, we choose two machine learning algorithms - LR and linear SVM - and test the output perturbation to assess data utility.

Fig. 2 shows that, according to the Pearson correlation coefficients, data correlation varies with the number of features. Data correlation generally decreases with a growing number



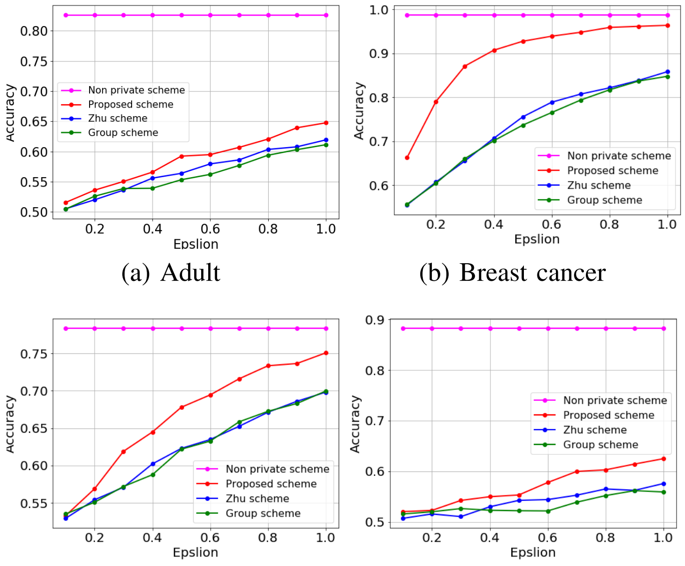
(a) Adult

(b) Breast Cancer

(c) Titanic

(d) Porto Seguro

Fig. 2: Data correlation for different number of features



(a) Adult

(b) Breast cancer

(c) Titanic

(d) Porto Seguro

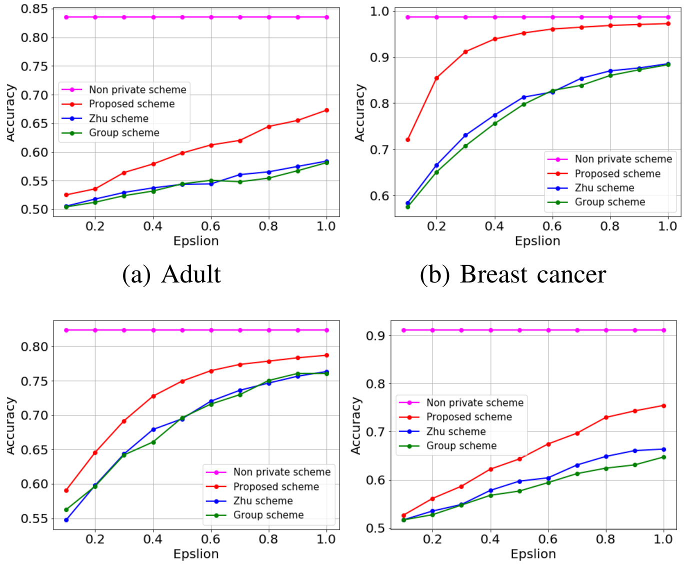
Fig. 3: Privacy-Accuracy trade-off in SVM for different datasets

of features but eventually stabilizes. For example, Figs. 2b and 2c show that data correlation become stable at 17 features with the Breast Cancer dataset and at 8 features with the Titanic dataset. This observation indicates that data correlation across the entire dataset can be reduced while preserving a suitable number of features for data analysis because more features means less correlation.

Table 3 shows the number of features in each dataset at different stages of the proposed scheme. It is noted that, Best feature set $\mathcal{B}$ will always contain more features than Adjusted feature set $\mathcal{A}$ and, as shown in the table, Adjusted feature set $\mathcal{A}$ have less data correlation than Best feature set $\mathcal{B}$, demonstrating that more features reduces correlation in a correlated dataset.

Figs. 3 and 4 show the performance of linear SVM and LR on different datasets with the four schemes. In most cases, LR have better accuracy than linear SVM. For example, Fig. 2a shows that when $\epsilon = 1$, LR have an accuracy of around 0.675 versus linear SVM's 0.645. Accuracy with the non-private scheme remains constant as the privacy budget increases and also performed better than the other schemes. This result demonstrates that imposing any form of privacy requirement on a dataset degrades data utility.

For the private schemes, the proposed scheme outperforms both the group and Zhu schemes in all circumstances. Figs. 3 and 4 show the level of improvement, especially Fig. 3b. $\epsilon = 1$, the proposed scheme scores an accuracy of around 0.97 compared to around 0.85 for the Zhu scheme. We attribute the improved performance of our scheme to the adjusted features. These additional features reduce data correlation but have little impact on the prediction results. Less data correlation means less noise needs to be added, which leads to better data utility. Other schemes do not reduce data correlation; they only consider how to accurately describe the data correlations, without considering that data correlation actually impedes accuracy.



(a) Adult

(b) Breast cancer

(c) Titanic

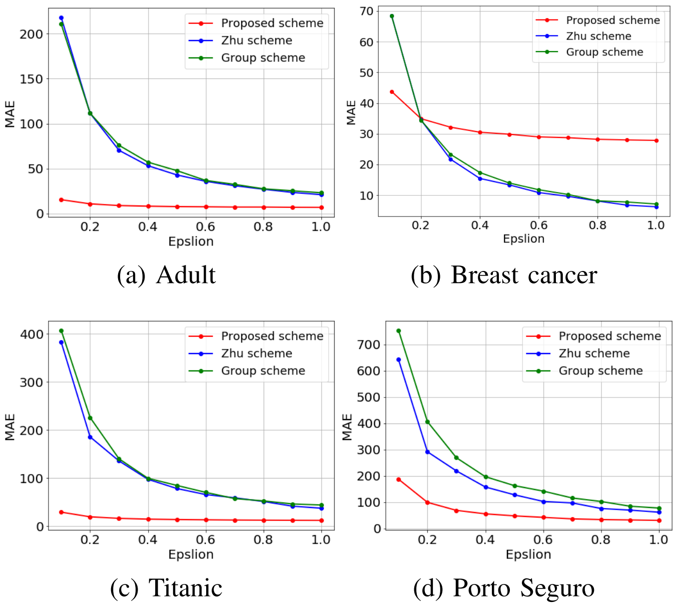
(d) Porto Seguro

Fig. 4: Privacy-Accuracy trade-off in LR for different datasets

Additionally, the group and Zhu schemes present closed curves with the first three datasets because the Pearson coefficient is set to a high-value $\theta_0 = 0.9$. This results in a similar correlated sensitivity for both schemes and, consequently, a similar level of noise is added. However, with the Porto Seguro dataset, we set the Pearson coefficient to $\theta_0 = 0.7$. Hence, there is a minor gap in performance. Also worthy of note is that the accuracy of prediction results varied for different datasets. This is due to the amount of data correlation in each dataset; higher correlation means more noise must be added, which reduces accuracy.

C. Experiments for data publishing

The second aim of our scheme is to improve utility for data publishing, which we evaluate with both count and mean queries. Mean absolute error (MAE) is used as the metric to assess both results, but different calculation formulas are defined to analyze the base results and the impact of varying the privacy budget. The accuracy of common queries is measured by MAE, which is given as,



(a) Adult (b) Breast cancer

(c) Titanic (d) Porto Seguro

Fig. 5: MAE performance for count queries

$$MAE = \frac{1}{|\mathcal{Q}|} \sum_{\mathcal{Q}_i \in \mathcal{Q}} |\hat{\mathcal{Q}}_i(x) - \mathcal{Q}_i(x)| \tag{18}$$

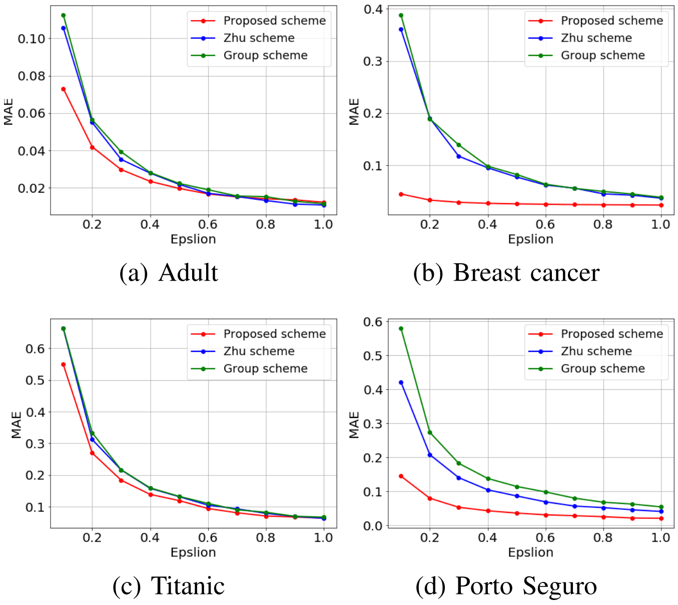
where $\mathcal{Q}_i(x)$ is the true aggregation result for one query, and $\hat{\mathcal{Q}}_i(x)$is the perturbed aggregation result calculates through different schemes. A low MAE indicates a low error and, thus, a better data utility.

To analyze how the proposed scheme performs with different privacy budgets, we also define a second MAE containing two components. One component measures the noise added due to correlated sensitivity, the other measures the errors introduced by adding the adjusted features. These features have an impact on a new query object that can emerge as errors when comparing the adjusted dataset to the original. This MAE is defined as

$$MAE = \frac{1}{|\mathcal{Q}|} \sum_{\mathcal{Q}_i \in \mathcal{Q}} |\hat{\mathcal{Q}}_i(x) - (\mathcal{Q}_i(x) - \mathcal{Q}_i^o(x))| \tag{19}$$

where $\hat{\mathcal{Q}}_i(x)$ and $\mathcal{Q}_i^o(x)$ are the true aggregation result on Best feature set $\mathcal{B}$ and Adjust feature set $\mathcal{A}$, respectively.

Fig. 5 shows the impact of varying privacy budgets on the performance of count queries in terms of MAE. With the proposed scheme, the MAE decreases as the privacy budget grows before stabilizing toward the end. This result demonstrates that a lower privacy requirement has better data utility. Moreover, the MAE for the proposed scheme is significantly smaller than the other schemes, which means that the proposed scheme does indeed improve data utility. For example, Figs. 5a and 5b at $\epsilon = 0.2$ show an MAE of around 18 for the Adult dataset and 17 for the Titanic dataset using our proposed scheme, whereas the group and Zhu schemes return an MAE of around 110 and 200 on these same datasets - an enormous increase over the proposed scheme. Again, we attribute these results to reduced data correlation after adding the adjusted features.



(a) Adult (b) Breast cancer

(c) Titanic (d) Porto Seguro

Fig. 6: MAE performance for mean queries

In terms of the other schemes, the MAE for the Zhu scheme is slightly lower than for the group scheme most of the time for the same reason as explained in the data analysis experiments. Moreover, the MAE for the Zhu scheme decreases faster as the privacy budget increased from 0.1 to 0.4 than when the budget increases from 0.4 to 1. This again shows that a higher privacy requirement creates a higher data utility cost.

The results of varying the privacy budgets with mean queries are similar, as shown in Fig. 6. However, the MAE are much smaller than for the count queries. This is because, after data normalization, the scale of data falls within $[-1, 1]$; therefore, each record has a similar mean value. As a result, the outcomes of mean queries are much smaller than for count queries. In addition, the MAE for our proposed scheme is not always better than the group or Zhu schemes - for example, when $\epsilon < 0.2$. This shows that adding the adjusted features can introduce additional errors. Hence, the quality of the query results in the proposed scheme depends on the type of queries and the dataset itself but, overall, our proposed scheme returns a lower MAE than the other schemes.

D. Discussion

The key to the CR-FS scheme is to reduce data correlation in the whole dataset, while maintaining a good utility for data analysis and data queries. We add differential private noise on two places: feature selection and data training and still can achieve desirable performance. This is because the fact that sensitivity of feature selection is smaller than 1, the sequence of feature importance will not change much. That is to say, there is a high probability that more important features are still more important and less important features are still less important. In this way, a higher probability that important features are kept for training and less important features are used to reduce data correlation.

For data analysis, we select features in the step 5 according to the accuracy of predicted results, thus the selected features can have less correlation across the whole dataset and achieve a desirable accuracy results. For data queries, the correlation in the whole dataset also reduced with the proposed CR-FS scheme. However, as we noted in the Figure 5 and 6, the MAE is not always better than other schemes. This is because the sensitivity is related the type of queries and dataset itself. Deleted or added features in the dataset can reduce the data correlation, which may bring in new error with regard to different queries.

VIII. Conclusion

In this paper, we identified the privacy issue of the data correlation in machine learning, which may result in more privacy loss than expected in industrial applications. We propose a novel feature selection scheme CR-FS to reduce data correlation with little compromise to data utility. The proposed CR-FS scheme includes steps that consider the accuracy of predicted results, the privacy preserving and the data correlation in the dataset. Our proposed algorithm strikes a better trade-off between data utility and privacy leaks for correlated datasets. The method's performance is evaluated via extensive experiments, and the results prove that our proposed CR-FS scheme provides better data utility for both data analysis and data queries compared to traditional schemes.

Acknowledgment

This work was supported in part by the Australian Research Council under Linkage Grant LP170100123.

References

[1] U.S. Shanthamallu, A. Spanias, C. Tepedelenlioglu and M. Stanley, "A brief survey of machine learning methods and their sensor and IoT applications," In *2017 8th International Conference on Information, Intelligence, Systems and Applications (IISA)*, pp. 1-8.

[2] I.A.T. Hashem, V. Chang, N.B. Anuar, K. Adewole, I. aqoob, A. Gani, E. Ahmed and H. Chiroma, "The role of big data in smart city," *International Journal of Information Management*, 36(5), pp.748-758.

[3] C. Yin, J. Xi, R. Sun and J. Wang, "Location privacy protection based on differential privacy strategy for big data in industrial internet of things," *IEEE Transactions on Industrial Informatics*, 2017, 14(8), pp.3628-3636.

[4] A. Solanas, C. Patsakis, M. Conti, I. Vlachos, V. Ramos, F. Falcone, O. Postolache, P. Perez-Martinez, R. Pietro, D. Perrea, "Smart health: a context-aware health paradigm within smart cities," *IEEE Communications Magazine*, vol. 52, no. 8, pp. 74–81.

[5] C.M. Benjamin, M. Fung, K. Wang, R. Chen and P.S. Yu, "Privacy-preserving data publishing: A survey of recent developments," *ACM Computing Surveys*, 2010, 42(4), pp.1-53.

[6] C. Dwork, 2006, "Differential privacy," in *ICALP*, pp. 1–12.

[7] M. Yang, T. Zhu, Y. Xiang and W. Zhou, 2018. "Density-based location preservation for mobile crowdsensing with differential privacy," *IEEE Access*, 2018, 6, pp.14779-14789.

[8] L. Lyu, K. Nandakumar, B. Rubinstein, J. Jin, J. Bedo, and M. Palaniswami, "PPFA: privacy preserving fog-enabled aggregation in smart grid," *IEEE Transactions on Industrial Informatics*, 2018, 14(8), pp.3733-3744.

[9] Y. Liu, W. Guo, C.I. Fan, L. Chang and C. Cheng, "A practical privacy-preserving data aggregation (3PDA) scheme for smart grid," *IEEE Transactions on Industrial Informatics*, 2019, 15(3), pp.1767-1774.

[10] D. Ye, T. Zhu, W. Zhou, and P.S. Yu, "Differentially Private Malicious Agent Avoidance in Multiagent Advising Learning," *IEEE transactions on cybernetics*, 2019, DOI:10.1109/TCYB.2019.2906574.

[11] K. Chaudhuri and C. Monteleoni, "Privacy-preserving logistic regression," In *Advances in neural information processing systems*, 2009, pp. 289-296.

[12] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate, "Differentially private empirical risk minimization," *Journal of Machine Learning Research*, 2011, 12:1069–1109.

[13] S. Song, K. Chaudhuri and A. D. Sarwate, "Stochastic gradient descent with differentially private updates," In *2013 IEEE Global Conference on Signal and Information Processing*, 2013, pp. 245-248.

[14] M. Abadi, A. Chu, I. Goodfellow, H.B. McMahan, I. Mironov, K. Talwar, and L. Zhang, "Deep learning with differential privacy," In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016, pp. 308-318.

[15] T. Zhu, P. Xiong, G. Li, W. Zhou and P.S. Yu, "Differentially private model publishing in cyber physical systems," *Future Generation Computer Systems*, 2018.

[16] D. Kifer and A. Machanavajjhala, "No free lunch in data privacy," In Proc. ACM SIGMOD Int. Conf. Manage. Data (SIGMOD), 2011, pp. 193–204.

[17] T. Zhu, P. Xiong, G. Li and W. Zhou, "Correlated differential privacy: Hiding information in non-iid data set," *IEEE Transactions on Information Forensics and Security*, 10(2), 2014, pp.229-242.

[18] T. Zhu, P. Xiong, G. Li and W. Zhou, "Answering differentially private queries for continual datasets release," *Future Generation Computer Systems*, 87, 2018, pp.816-827.

[19] B. Yang, I. Sato, and H. Nakagawa, "Bayesian Differential Privacy on Correlated Data," *ACM SIGMOD International Conference on Management of Data*, 2015:747-762.

[20] J. Chen, H. Ma, D. Zhao, and L. Liu, "Correlated Differential Privacy Protection for Mobile Crowdsensing," in *IEEE Transactions on Big Data*

[21] Y. Cao, M. Yoshikawa, Y. Xiao and L. Xiong, "Quantifying Differential Privacy in Continuous Data Release under Temporal Correlations," in *IEEE Transactions on Knowledge and Data Engineering*.

[22] D. Kifer and A. Machanavajjhala, "Pufferfish: A framework for mathematical privacy definitions," *ACM Trans. Database Syst.*, Jan. 2014, 39(1):3:1–3:36.

[23] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers and Electrical Engineering,* 2014, 40(1):16-28.

[24] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Foundations and Trends in Theoretical Compututer Science*, Aug. 2014, vol. 9, pp. 211–407.

[25] H. Liu and L. Yu, "Toward integrating feature selection algorithms for classification and clustering," *IEEE Transactions on Knowledge and Data Engineering*, 2015, pp.491-502.

[26] T. Zhu, P. Xiong, G. Li, W. Zhou and P.S. Yu, "Differentially private data publishing and analysis: A survey," *IEEE Transactions on Knowledge and Data Engineering*, 2017, 29(8), pp.1619-1638.

[27] A. Asuncion and D.J. Newman, UCI Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences, 2007. Available: http://www.ics.uci.edu/∼mlearn/MLRepository.html.

[28] A. Asuncion and D.J. Newman, UCI Machine Learning Repository. University of California, Irvine, School of Information and Computer Sciences, 2007. Available: https://www.kaggle.com/rohitjain2086/breast-cancer-dataset-prediction/data

[29] https://www.kaggle.com/c/titanic

[30] https://www.kaggle.com/c/porto-seguro-safe-driver-prediction

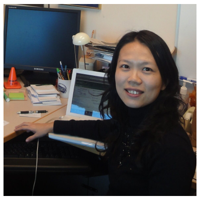
[31] R. Chen, B.C. Fung, P.S. Yu and B.C. Desai, "Correlated network data publication via differential privacy," *The International Journal on Very Large Data Bases,* 2014, 23(4), pp.653-676.

Tao Zhang received the B.S. degree and M.S. degree from the Information Engineering School, Nanchang University, China, in 2015 and 2018, respectively.

Currently, he works towards his Ph.D degree with the school of Computer Science in the University of Technology Sydney, Australia. His research interests include privacy preserving, AI fairness, data mining and machine learning.

Tianqing Zhu received her BEng and MEng degrees from Wuhan University, China, in 2000 and 2004, respectively, and a PhD degree from Deakin University in Computer Science, Australia, in 2014. Dr Tianqing Zhu is currently a senior lecturer in the school of Computer Science in the University of Technology Sydney, Australia. Before that, she was a lecture in the School of Information Technology, Deakin University, Australia, from 2014 to 2018. Her research interests include privacy preserving, data mining and network security.

Ping Xiong received his BEng degree from LanZhou Jiaotong University, China in 1997. He received his MEng and PhD degrees from Wuhan University, China, in 2002 and 2005, respectively. He is currently the professor of School of Information and Security Engineering, Zhongnan University of Economics and Law, China. His research interests are network security, data mining and privacy preservation.



Huan Huo received the B.Eng and Ph.D. degrees from Northeastern University, China in 2002 and 2007, both in Computer Science and Technology. From 2012 to 2014, Angela HUO taught at the Department of Computer Information System, the University of the Fraser Valley in Canada, and did collaborative research in the University of Waterloo as a visiting scholar for one year. Since 2018, she has been a senior lecture in the school of software at the University of Technology Sydney, Australia. Her research interests include data stream management technology, advanced data analysis, and data-driven cybersecurity.

Zahir Tari is a full professor in distributed systems at RMIT University, Australia. He received his Ph.D. degree in computer science from the University of Grenoble, France, in 1989. His expertise is in the areas of system performance (e.g., cloud, IoT) as well as system security (e.g., SCADA, cloud). He was an Associate Editor of IEEE Transactions on Computers, IEEE Transactions on Parallel and Distributed Systems, and IEEE Magazine on Cloud Computing.

Wanlei Zhou received the B.Eng and M.Eng degrees from Harbin Institute of Technology, Harbin, China in 1982 and 1984, respectively, and the PhD degree from The Australian National University, Canberra, Australia, in 1991, all in Computer Science and Engineering. He also received a DSc degree (a higher Doctorate degree) from Deakin University in 2002. He is currently the Head of School of Software in University of Technology Sydney (UTS). Before joining UTS, Professor Zhou held the positions of Alfred Deakin Professor, Chair of Information Technology, and Associate Dean (International Research Engagement) of Faculty of Science, Engineering and Built Environment, Deakin University. His research interests include security and privacy, bioinformatics, and e-learning. Professor Zhou has published more than 400 papers in refereed international journals and refereed international conferences proceedings, including many articles in IEEE transactions and journals.