Journal of Computers Vol. 34 No. 1, February 2023, pp. 157-173
doi: 10.53106/199115992023023401012

# Differentially Private Feature Selection Based on Dynamic Relevance for Correlated Data

Chunxia Wang, Qiuyu Zhang*, Yan Yan

School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China
wcxia1985@163.com, {zhangqy, yanyan}@lut.edu.cn

Received 11 April 2022; Revised 5 August 2022; Accepted 11 August 2022

**Abstract.** Traditional feature selection methods are only concerned with high relevance between selected features and classes and low redundancy among features, ignoring their interrelations which partly weak classification performance. This paper developed a dynamic relevance strategy to measure the dependency among them, where the relevance of each candidate feature is updated dynamically when a new feature is selected. Protecting sensitive information has become an important issue when executing feature selection. However, existing differentially private machine learning algorithms have seldom considered the impact of data correlation, which may cause more privacy leakage than expected. Therefore, the paper proposed a differentially private feature selection based on dynamic relevance measure, namely DPFSDR. Firstly, as a correlation analysis technique, the weighted undirected graph model is constructed via the correlated degree, which can reduce the dataset's dimension and correlated sensitivity. Secondly, as a feature selection criterion, *F*-score with differential privacy is adopted to measure the feature importance of each feature. Finally, to evaluate the effectiveness of feature selection, differentially private SVM combined with dynamic relevance measure is utilized to choose features. Experimental results show that the proposed DPFSDR algorithm can effectively obtain the optimal feature subset, and improve data utility while preserving data privacy.

**Keywords:** feature selection, correlated data, differential privacy, associated attribute, dynamic relevance measure

## 1 Introduction

With the rapid development of information technology, privacy preservation in data mining increasingly has become an important issue. The main aim is to protect the sensitive information that data owner is reluctant to disclose, which has been growingly concerned in financial records, medical records, web search histories, and social networks. Because of the advancements in the internet throughput technologies, the collected data of individuals by the online systems are mostly high dimensional, which will make the data mining tasks more difficult, and cause the curse of dimensionality.

Feature selection is known to be an essential pre-processing technology in data mining. Traditionally, feature selection can not only reduce data dimension by eliminating irrelevant and redundant features as many as possible, but also lower computation consumption and improve classification performance. Apart from the identification of irrelevant and redundant features, an important but commonly ignored issue is feature interaction. Interacting features are those that appear to be irrelevant to the class individually, but when combined with other features, they may be highly correlated to the class.

Although some recent research has pointed out the effect of feature interaction on classification performance, there is little work on explicit treatment of feature interaction. A linear feature selection method, namely, the dynamic change of selected feature with the class (DCSF) was proposed [1], but it failed to take into account the interaction among candidate features, selected features, and classes in the feature selection process, resulting in the decline of classification performance.

Unfortunately, the process of feature selection has the potential to reveal private information, but the existing feature selection methods are seldom concerned with the issues of privacy loss. Moreover, the collected data may contain some associated attributes and correlated records due to temporal correlation or user correlation, which will further increase the risk of privacy leaks. Therefore, it is essential to preserve the private information of correlated data during feature selection.

* Corresponding Author

As a popular technique for privacy preservation, differential privacy proposed by Dwork has attracted considerable attention due to its rigorous mathematical framework and independent background knowledge [2]. Current studies on differential privacy mainly focus on data independence, but the fact is not usually true, the collected data usually have complex correlation rather than complete dependency. Kifer et al. confirmed that the correlated data might reveal more privacy information than expected [3]. Thus, it is greatly necessary to consider the impact of the data correlation when designing differentially private machine learning algorithms. At present, data correlation mainly involves temporal correlation [4], trajectory correlation [5], and attribute association [6-7]. This paper focuses on the loss of privacy caused by attribute association.

Moreover, the correlated sensitivity of a dataset is commonly related to the number of features. This means more features may have a lower correlated sensitivity and vice versa. Dimensionality reduction and important feature selection play a dominant role in improving the machine learning classification performance, but dimensional reduction causes a further increment of sensitivity due to data correlation, which degrades the data utility.

Based on the above analysis, the main challenges are as follows.

(1) Lack of feature interaction during feature selection may cause performance to decline.

(2) Correlate data will make more privacy leak, so it is crucial to protect the private information of correlated data in the process of feature selection.

(3) Dimensionality reduction further increases correlated sensitivity due to data correlation, which reduces the data utility.

These challenges imply that a novel mechanism for differentially private feature selection for correlated data is in high demand. With respect to the first challenge, to measure feature interaction among selected features, candidate features, and classes, a dynamic relevance strategy is explored, where the relevance of each candidate feature is dynamically modified when a new feature is added to the selected feature set. For the second challenge, to reduce data correlation, the weighted undirected graph model is used to filter the associated attributes and correlated records. As regards the third challenge, the correlation variation caused by dimensionality changes is analyzed, and the number of features selected is determined by private SVM combined with Sequential Forward Floating Search (SFFS).

In summary, the major contributions of this paper are as follows.

(1) A differentially private feature selection based on dynamic relevance (DPFSDR) is proposed in this paper. It can select features privately by calculating the feature importance of every feature while concerning the dependency with dynamic relevance measure among selected features with dynamic relevance measure, candidate features, and classes, thus maintaining a desirable data utility.

(2) A correlation analysis technique is used to reduce the dataset's dimensionality and correlated sensitivity when implementing differential private machine learning algorithms, and thus improve data utility.

(3) To evaluate the performance of the proposed feature selection scheme, the differentially private SVM is designed. The experimental results from four different UCI datasets demonstrate that the proposal can achieve a better trade-off between data privacy and data utility than existing methods.

The rest of this paper is organized as follows. Section 2 summarizes the previous work in differentially private feature selection and correlated differential privacy. Section 3 introduces some preliminaries and basic definitions. Section 4 describes the proposed method. Section 5 demonstrates the experiment results and analysis in detail. Finally, Section 6 gives the conclusions.

## 2 Related Work

### 2.1 Differentially Private Feature Selection

Differentially private feature selection mainly solves the problems of reducing the high dimensionality of the dataset privately and thus improving the data utility. In related literature, Li et al. [8] proposed a local learning-based feature weighted framework, and output perturbation and objective perturbation were adopted to improve privacy preserving property for local learning-based feature selection algorithm, where logistic loss with L2-regularizer was utilized to design the evaluation criterion of feature selection. Le et al. [9] developed a private Evaporative Cooling algorithm, which used Relief-F for feature selection and random forest for classification with an exponential differential privacy mechanism while avoiding over-fitting caused by the feature number being far larger than the sample number. In order to query data aggregation from high-dimensional data sets under differential privacy protection, He et al. [10] proposed a differentially private feature selection method based on a data sampling process with a K-D tree. It returned differentially private data aggregates from a low-dimensional dataset, and a two-stage noise injection was used to satisfy the trade-off between privacy and utility of data aggregates. To solve the privacy problem caused by automatic selection techniques based on MI ranking, Srivastava et al. [11] proposed a Distributed Differentially Private Mutual Information (DDP-MI), as a privacy-safe batch MI, and was used in some scenarios such as feature selection, segmentation, ranking and query expansion. Moreover, the distributed implementation provided a strong guarantee against various privacy attacks and substantially improved the efficiency of MI calculations. Previous works focused on the adding of privacy preservation to the single feature selection algorithm, Liu et al. [12] proposed a differentially private ensemble feature selection to improve the classification accuracy and stability of feature selection.

Many researchers have accumulated diverse results on privacy protection. However, these existing studies paid little attention to data correlation issues.

### 2.2 Correlated Differential Privacy

Differential privacy provides a rigorous mathematical method for defining indistinguishability to protect privacy and ensures that adding or removing any single record does not change the analysis results. Previous studies have shown that the correlated data give rise to more privacy loss problems. In practice, completely independent data rarely exists. There have been two types of differential privacy mechanisms for correlated data. One is model-based mechanisms, where the correlation model is built and noise conforming to the model is generated to output perturbation. The other is to optimize the sensitivity function in terms of the number of correlated records or correlation coefficient matrix.

On the one hand, in the model-based mechanisms, Kifer et al. [3] conceived the Pufferfish framework, which analyzed the impact of data correlation on differential privacy in detail, but it failed to satisfy differential privacy. Inspired by the Pufferfish framework, He et al. [13] proposed the Blowfish model, which balanced privacy loss and data utility with specifying secrets and constraints. Another privacy definition of Pufferfish was Bayesian differential privacy [14], which introduced the Gaussian correlation model to describe the structure of data correlation, and analyzed the privacy level of different perturbation algorithms based on this model. Because of the lack of an appropriate privacy mechanism for Pufferfish, Song et al. [15] presented the Wasserstein mechanism, which was applied to any Pufferfish instantiation. Liu et al. [16] utilized the hide Markov model to express trajectory correlation, and measured sensitivity with the Markov model, which determined the scale of noise. Liao et al. [17] combined game theory and the Markov model to achieve the trade-off between data privacy and data utility. Ju et al. [18] designed a correlation-based privacy protection scheme for social graph data. In order to add adequate noise to the query results, the data sensitivity between the original graph and the randomized graph was recalculated according to the data correlation. These above mechanisms were based on conditional probability and generated Laplace noise whose joint probability density complies with a given model, such as the Gaussian model or Bayesian model, so there was some strict restriction on original data structure in model-based mechanisms.

On the other hand, in the study of optimizing sensitivity function, to avoid large-scale noise caused by excessive correlated data, Liu et al. [19] introduced dependent differential privacy (DDP) framework incorporating probabilistic dependence relationship between tuples in the statistical database. Besides, the dependent perturbation mechanism (DPM) was used to achieve the privacy guarantees in DDP. Similarly, the refined sensitivity function through the dependence coefficient obtained less noise. Lv et al. [20] proposed k-CRDP and r-CBDP models to protect the privacy of correlated data in big data. Firstly, r-CBDP combined the Maximum Information Coefficient (MIC) and machine learning algorithms to determine dependencies between data, accurately calculated the sensitivity, then divided the big data set into several independent blocks, and implemented k-CRDP for the blocks to achieve correlated differential privacy of big data. Liang et al. [21] generalized the Pufferfish model and designed a privacy leakage computation model (PLCM) as a quantitative analysis of the maximum privacy leakage caused by temporal correlations. Almadhoun et al. [6] calculated the sensitivity function with the probability dependence between data tuples and the "adjustable" value, which helped to decrease the noise magnitude and provided a rigorous privacy guarantee. Zhu et al. [22] introduced the correlation coefficient matrix to describe the data correlation, the correlation coefficient was used as the weight of global sensitivity to calculate the sensitivity function, and feature selection was implemented to lower data correlation. However, Zhu et al. [22] overlooked the dependency among selected features, candidate features, and classes. Moreover, adding or deleting features from the adjusted feature set decreased data correlation in the whole dataset, but such adjustment scheme failed to reflect the influence of modifying a record on the change of query results. Thus, these above schemes effectively reduce the noise for privacy protection by optimizing sensitivity function but still have a poor utility when dealing with large-scale correlated records.

## 3 Preliminaries

### 3.1 Differential Privacy

Differential privacy is a rigorous privacy model that does not involve any assumptions regarding the background knowledge of adversaries and guarantees that the change of any single record in the dataset does not significantly shift the output distribution [2]. In brief, given two datasets $D$ and $D^i$ that contain a set of records, they are neighboring if having the same cardinality but differ in only one record. Let $r_i$ be that record, then $D$ represents the dataset with $r_i$ and $D^i$ represent the dataset with $r_i$ deleted from $D$. A query $f$ is a function that maps the record $r \in D$ into outputs $f(D) \in S$, where $S$ is the whole set of outputs. The formal notion of differential privacy is shown as follows.

**Definition 1.** $\varepsilon$-Differential Privacy. A random algorithm $A$ satisfies $\varepsilon$-differential privacy if for any pair of $D$ and $D^i$, and for any possible outcome $f(D) \in S$, $A$ will be satisfy $\varepsilon$-Differential Privacy, if

$$P[A(D) \in S] \leq \exp(\varepsilon) \times P[A(D^i) \in S] \ , \tag{1}$$

where $\varepsilon$ refers to the privacy budget that is used to tune the privacy level of the mechanism $A$. The lower $\varepsilon$ means the higher privacy level.

**Definition 2.** Global Sensitivity. For any query $f : D \rightarrow \mathbb{R}$, the sensitivity of $f$ is defined as

$$\Delta f = \max_{D,D^i} \left\| f(D) - f(D^i) \right\|_1 \ , \tag{2}$$

where $D$ and $D^i$ are neighboring datasets which is only related to the type the query $f$. Global sensitivity measures the maximal difference on the result of query $f$ when removing one record from the dataset $D$. The larger $\Delta f$ is, the greater will be the addition of noise required to mask the effect of all the records of the datasets [23].

**Definition 3.** Laplace Mechanism. For any query $f : D \rightarrow \mathbb{R}$, Laplace mechanism satisfies $\varepsilon$-differential privacy if

$$A(D) = f(D) + lap(\frac{\Delta f}{\varepsilon}) \ , \tag{3}$$

where $lap(\cdot)$ denotes Laplace noise drawn from a Laplace distribution with probability density function $p(x \mid \lambda) \frac{1}{2\lambda} e^{-|x|/\lambda}$, and $\lambda$ depends on the sensitivity and the privacy budget.

**Theorem 1.** Sequence composition [24]. Suppose random algorithm $A_1, A_2, ..., A_n$, and their privacy budgets are $\varepsilon_1, \varepsilon_2, ..., \varepsilon_n$. As for the same dataset $D$, $A(A_1(D), A_2(D), ..., A_n(D))$, the combination algorithms of $A_1, A_2, ...,$ $A_n$ on $D$, satisfies $\varepsilon$-differential privacy and $\varepsilon = \sum_{i=1}^{n} \varepsilon_i$.

### 3.2 Correlated Sensitivity

For a query, real-world datasets often cover some records partially correlated, that is to say, modifying one record has a probability to change other correlated records. Correlated sensitivity is introduced to measure how much effect on other records when modifying one record.

**Definition 4.** Correlated sensitivity [22]. For a query $f$, the correlated sensitivity $\Delta CS_Q$ is based on the correlated degree and the number of correlated records, which is defined as,

$$\Delta CS_Q = \max_{i \in Q} \sum_{j=0}^{n} | \theta_{ij} | (\| f(D^j) - f(D^{-j}) \|_1) \ , \tag{4}$$

where $Q$ is a record set of all records responding to query $f$, $\theta_{ij}$ 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 covers all the sensitivity of records with the query $f$. When a query just contains the independent or weak correlated record, the correlated sensitivity will not generate additional noise. For any query, the perturbed answer is adjusted with Equation (5).

$$\hat{f}(D) = f(D) + \text{Laplace}(\frac{\Delta CS_Q}{\varepsilon}) \ . \tag{5}$$

## 4 The Proposed Method

The proposed feature selection scheme mainly involves three steps, shown as follows.

**Step 1:** Design the weighted undirected graph of associated attributes, and eliminate the attributes whose correlated degree is higher than the given threshold, as described in Section 4.1;

**Step 2:** Compute the importance of every feature via differentially private *F*-score, and sorted these features in descending order according to their feature importance, as described in Section 4.2;

**Step 3:** Implement DPFSDR with SFFS strategy, as described in Section 4.3. (**DPFSDR Algorithm Description.**)

### 4.1 The Weighted Undirected Graph Model

Given the dataset $D$, the attribute set is represented with $X = \{ X_1, X_2, ..., X_i, ..., X_m \}( 1 \le i \le m )$, where $m$ denotes the number of dimension of dataset $D$. The Pearson correlation coefficient is an efficient way to discover associated attributes and correlated records in dataset.

**Definition 5.** Correlated degree. Suppose two attributes $X_i$ and $X_j$, then correlated degree between $X_i$ and $X_j$ can be described by Pearson correlation coefficient as follows,

$$\theta_{ij} = p(X_i, X_j) = \frac{E[(X_i - u_{X_i})(X_j - u_{X_j})]}{\sigma_{X_i}\sigma_{X_j}} \ , \tag{6}$$

where $u_{X_i}, u_{X_j}$ are mean of $X_i$ and $X_j$ respectively, $\sigma_{X_i}, \sigma_{X_j}$ are covariance of $X_i$ and $X_j$ respectively.

**Corollary 1.** If $\theta_{ij} = \theta_{ji} = 0$, it indicates no relationship between $X_i$ and $X_j$; if $| \theta_{ij} | = 1$, $X_i$ and $X_j$ are fully correlated; if $0 < \theta_{ij} < 1$, then have a positive correlated; if $-1 < \theta_{ij} < 0$, they have a negative correlated.

From correlated data analysis, a weighted undirected graph of associated attributes is constructed in this paper, where the vertex set represents the attribute set, the edge set represents the correlation between attributes, and the weight of edge $\theta_{ij}$ represents the correlated degree between $X_i$ and $X_j$. The weighted undirected graph is formally described by adjacency matrix $M$.

**Definition 6.** Correlated Degree Adjacency matrix $M$. It is possible to list all relationships between attributes,

$$M[i][j] = \begin{cases} \theta_{ij}, & \theta_{ij} < \delta \\ 0, & \theta_{ij} \ge \delta \end{cases} \quad 1 \le i, j \le m \ , \tag{7}$$

where, $\theta_{ij}$ is the correlated degree between $X_i$ and $X_j$, the threshold $\delta$ is used to filter the attributes with the higher correlated degree, and $M[i][j]=0$ implies they have no relationship at all.

Here are two properties of $M$,

(1) It is symmetrical with $\theta_{ij}=\theta_{ji}$.

(2) The elements on the diagonal are equal to 0.

The Correlated Degree Adjacency matrix $M$ is generated in detail as Algorithm 1.

**Algorithm 1.** Generate correlated degree adjacent matrix $M$

```
Input: dataset D, the attribute set $X = \{X_1, X_2, ..., X_i, ..., X_m\}$, the threshold $\delta$
Output: Adjacent Matrix $M$
for $i=1$ to $m$
    for $j=1$ to $m$
        $M[i][j]=0$
    end for
end for
for $i=1$ to $m$
    for $j=1$ to $m$
        Compute $\theta_{ij}$
        if $\theta_{ij} >= \delta$ then
            $\theta_{ij}=0$
        end if
        $M[i][j]=\theta_{ij}$
    end for
end for
Output $M$
```

From Algorithm 1, the computational complexity is $O(m^2)$, and its elements on the diagonal are equal to zero. So, with the idea of compression storage, the elements in $M$ is stored with $m(m-1)/2$ units. Besides, lower associated attributes are selected according to the given threshold $\delta$.

### 4.2 Differentially Private *F*-score Feature Selection

**Definition 7.** *F*-score [25]. Given training dataset $x_k \in R^m$, $k=1,2,...,n$, and number of datasets is $l$, if the number of $i$-th dataset is $n_j$, $j=1,2,...,l$, then the *F*-score of $i$-th feature is

$$F_i = \frac{\sum_{j=1}^{l}(\overline{x}_i^j - \overline{x}_i)^2}{\sum_{j=1}^{l}\frac{1}{n_j-1}\sum_{k=1}^{n_j}(x_{k,i}^j - \overline{x}_i^j)^2}, \tag{8}$$

where, $\overline{x}_i$, $x_i^j$ are average of the $i$-th feature of whole data set, and the $j$-th data set respectively, $x_{k,i}^j$ is the $i$-th feature of $k$-th record in $j$-th dataset. The numerator represents the discrimination between each dataset, and denominator denotes the one within each of dataset. Thus, the larger the *F*-score is, the more likely discriminative this feature is. *F*-score is used to calculate the importance of each feature, namely fim, and neighboring data are obtained when deleting record $r_i$.

**Definition 8.** Feature Importance. Given the dataset $D$, the feature importance of $i$-th feature is defined as

$$\mathrm{fim}_i = F_i / \sum_i F_i \ , \tag{9}$$

where $F_i$ is calculated by Equation (8). The larger $\mathrm{fim}_i$ is, the more likely discriminative this feature is, and the more important the feature to the class is.

**Definition 9.** Record Sensitivity of Feature Importance. For a query $f$, the record sensitivity of feature importance of record $r_i$ is denoted as

$$\Delta\mathrm{fim}_i = \| \mathrm{fim}^i_{\max} - \mathrm{fim}^i_{\min} \|_1 \ , \tag{10}$$

where $\mathrm{fim}^i_{\max}$ and $\mathrm{fim}^i_{\min}$ represent maximum and minimum of feature importance for record $r_i$ respectively.

**Definition 10.** Sensitivity of Feature Importance. For a query $f$, the sensitivity of feature importance is determined by the maximal record sensitivity of feature importance,

$$\Delta\mathrm{fim}_Q = \max_{i \in Q}(\Delta\mathrm{fim}_i) \ , \tag{11}$$

where $Q$ is a set of records related to a query $f$. It is easy to know $\Delta\mathrm{fim}_Q \le 1$, since the range of feature importance is from 0 to 1. Besides, Laplace mechanism is used to added noise to the feature importance, and the perturbed feature importance is defined as

$$\hat{\mathrm{fim}}_i = \mathrm{fim}_i + \mathrm{Lap}(\frac{\Delta\mathrm{fim}_Q}{\varepsilon_1}) \ . \tag{12}$$

Then the perturbed feature importance is normalized as

$$\mathrm{fim}'_i = \hat{\mathrm{fim}}_i / \sum_{i=1}^{m} \hat{\mathrm{fim}}_i \ . \tag{13}$$

According above descriptions, Feature selection with Differential Private *F*-score is shown as follows.

**Algorithm 2.** Differentially private *F*-score feature selection

```
Input: dataset $D$, the privacy budget $\varepsilon_1$
Output: The new sequence of feature importance $F_{cd}$
  1. Calculate $F_i$ for each feature according to Equation (8)
  2. for $i = 1$ to $m$
  3.        Normalize $\mathrm{fim}_i = F_i / \sum_i F_i$
  4.        Perturb $\hat{\mathrm{fim}}_i = \mathrm{fim}_i + \mathrm{Lap}(\frac{\Delta\mathrm{fim}_Q}{\varepsilon_1})$
  5. Normalize $\mathrm{fim}'_i = \hat{\mathrm{fim}}_i / \sum_{i=1}^{m} \hat{\mathrm{fim}}_i$, $i = 1, ..., m$
  6. Sorted $\mathrm{fim}'_i$ in descending order, denotes $f_1 > f_2 > ... > f_m$
  7. Output the new sequence of feature importance $F_{cd} = \{f_1, f_2, ..., f_i, ..., f_m\}$
```

### 4.3 Proposed DPFSDR Method

**Dynamic Relevance Measure.** Although the feature importance can measure effectively the discrimination of the feature to class, it fails to evaluate the interrelation among selected features, candidate features, and classes. So, a dynamic relevance measure is proposed in this paper, where the relevance of each feature in the candidate feature set is dynamically updated through selected features and classes. The average redundancy between a candidate feature and the selected feature subset is defined as

$$R_i = \frac{\sum_{j=1}^{|F_{sel}|} |\theta_{ij}|}{|F_{sel}|} , \tag{14}$$

where $F_{sel}$ denotes selected feature subset, $|F_{sel}|$ denotes the number of selected features in $F_{sel}$, the correlated degree $\theta_{ij}$ between the candidate feature $X_i$ and the $j$ -th feature in $F_{sel}$ is calculated by Pearson correlation coefficient.

Combining the average redundancy and the importance of each feature in candidate feature subset, the dynamic relevance measure is represented as

$$DR_i = \frac{\mathrm{fim}_i^{'}}{R_i} , \tag{15}$$

where $DR_i$ denotes the dynamic relevance of $i$ -th feature in candidate feature subset. The larger $\mathrm{fim}_i^{'}$ and the smaller $R_i$ is, the more likely discriminative to classes and lower redundancy with selected features is.

**DPFSDR Algorithm Description.** The DPFSDR procedure is implemented with SFFS as the following:

**Step 1:** Initialize the selected feature subset empty and the candidate feature subset with all features, ranking in descending order according to their feature importance;

**Step 2:** Select the top one feature from the candidate feature subset and add it to the selected feature subset;

**Step 3:** Build the predictor model to classify the training subset of samples according to the current selected feature subset, get the classification accuracy of the training subset, and then use the output perturbation test;

**Step 4:** Compute the dynamic relevance of each feature in the candidate feature subset, and sort them in descending order;

**Step 5:** Select the top one feature from the candidate feature subset and copy it to the selected feature subset;

**Step 6:** Build the predictor model to classify the training subset of samples according to the current selected feature subset, get the classification accuracy of the training subset, and then use the output perturbation test;

**Step 7:** If the prediction accuracy is improved, eliminate the feature from the candidate feature subset that has just been copied to the selected feature subset, and then go to Step 4;

**Step 8:** Else eliminate the feature from the selected feature subset that has just been copied to the selected feature subset, select the next top one feature from the candidate feature subset and copy it to the selected feature subset, and then go to Step 6;

**Step 9:** Execute the procedure until the candidate feature subset is empty or all features in the candidate feature subset have been processed but they have no change to the prediction accuracy.

The specific DPFSDR procedure is shown in Algorithm 3.

**Algorithm 3.** DPFSDR algorithm

**Input:** The privacy budget $\varepsilon_2$ , the candidate feature subset $F_{cd}$

**Output:** The optimal feature subset $F_{sel}$

```
1.   Initialize $F_{sel} = \phi$
2.   $f_1$ = arg max($F_{cd}$), $f_1 \in F_{cd}$
3.   $F_{sel} = \{f_1\}, F_{cd} = F_{cd} - \{f_1\}$
4.   For $F_{sel}$ , use linear SVM classifier to obtain classification hyperplane ($w$, $b$)
5.   Perturb $w : w' = w + \text{Laplace}(\frac{\Delta CS_Q}{\varepsilon_2})$
6.   Get the prediction results $Acc$ according to $w'$
7.   Do while $F_{cd} \neq \phi$
8.       Flag=False
9.       Calculate $DR_i$ for each feature in $F_{cd}$,
         and sort them in descending order according to Equation (15)
10.      Select the top one feature $f_p$ from $F_{cd}$, $f_p$ = GetFirstElem($F_{cd}$) , $f_p \in F_{cd}$
11.      Do While $f_p \neq \phi$
12.          $F_{sel}^* = F_{sel} + \{f_p\}$
13.          For $F_{sel}^*$, use linear SVM classifier to obtain classification hyperplane ($w$, $b$)
14.          Perturb $w : w' = w + \text{Laplace}(\frac{\Delta CS_Q}{\varepsilon_2})$
15.          Get the prediction results $Acc^*$ according to $w'$
16.          if $Acc^* > Acc$ then
17.              Flag=True
18.              $Acc = Acc^*$
19.              $F_{sel} = F_{sel}^*$
20.              $F_{cd} = F_{cd} - \{f_p\}$
21.              goto 7
22.          else
23.              $f_p$ = GetFirstElem($F_{cd} - \{f_p\}$)
24.              goto 11
25.          end if
26.          if Flag==False then
27.              goto 31
28.          end if
29.      end Do
30.  end Do
31.  Output the optimal feature subset $F_{sel}$
```

To validate the effectiveness of the DPFSDR algorithm, it is compared with the proposed Differential Private Feature Selection (DPFS) algorithm, where SFFS is used to conduct the feature selection process, and the feature with highest feature importance in the candidate feature subset is selected and added to the selected feature subset, then evaluate the classification performance with output perturbation. If the classification accuracy is not improved, then eliminate the feature that has been added to the selected feature subset. The DPFS procedure is executed until all features in the candidate feature subset have been processed. DPFS is described in detail as Algorithm 4.

**Algorithm 4.** DPFS algorithm

**Input:** dataset $D$, the privacy budget $\varepsilon_2$, the candidate feature subset $F_{cd}$

**Output:** The optimal feature subset $F_{sel}$

```
Initialize $F_{sel} = \phi$
$f_1 = \arg\max(F_{cd}),\ f_1 \in F_{cd}$
$F_{sel} = \{f_1\},\ F_{cd} = F_{cd} - \{f_1\}$
For $F_{sel}$, use linear SVM classifier to obtain classification hyperplane $(w, b)$
Perturb $w : w' = w + \text{Laplace}(\frac{\Delta CS_Q}{\varepsilon_2})$
Get the prediction results $Acc$ according to $w'$
Do while $F_{cd} \neq \phi$
    Select the top feature $f_p = \text{GetFirstElem}(F_{cd}),\ f_p \in F_{cd}$
    $F_{sel}^* = F_{sel} + \{f_p\}$
    For $F_{sel}^*$, use linear SVM to obtain classification hyperplane $(w, b)$
    Perturb $w : w' = w + \text{Laplace}(\frac{\Delta CS_Q}{\varepsilon_2})$
    Get the prediction results $Acc^*$ according to $w'$
    if $Acc^* > Acc$ then
        $Acc = Acc^*$
        $F_{sel} = F_{sel}^*$
    end if
    $F_{cd} = F_{cd} - \{f_p\}$
end Do
Output the optimal feature subset $F_{sel}$
```

### 4.4 Privacy Analysis

**Theorem 2.** The proposed DPFSDR scheme satisfies $\varepsilon$-differential privacy.

To prove the DPFSDR method satisfies $\varepsilon$-differential privacy, this paper first analyzes which steps consume the privacy budget in the DPFSDR scheme. From Algorithm 2 and Algorithm 3, the dataset is used in two places: 1) feature selection procedure with differentially private *F*-score, 2) data training stage. To protect data privacy, noise is added to these two places. The total privacy budget $\varepsilon$ is divided into $\varepsilon_1$ and $\varepsilon_2$, and allocated to these two places respectively. First, the privacy budget $\varepsilon_1$ is analyzed in the process of feature selection.

**Lemma 1.** The feature selection procedure with differentially private *F*-score satisfies $\varepsilon_1$-differential privacy.

The neighboring datasets $D$ and $D^i$ are obtained by deleting the record $r_i$ from the dataset $D$, and $f_1(\cdot)$ is the query of feature selection. $p_x(z)$ and $p_y(z)$ are the probability density function as following:

$$A_1(x, f_1(\cdot), \varepsilon_1) = f_1(x) + \mathrm{Lap}(\frac{\Delta \mathrm{fim}_Q}{\varepsilon_1}) \ . \tag{16}$$

Suppose $x$ and $y$ be neighboring datasets, the random points $z \in \mathbb{R}$ are compared and ratio of two probability density can be denote as

$$\begin{aligned}
\frac{p_x(z)}{p_y(z)} &= \prod_{i=1}^{m}(\frac{\exp(-\frac{\varepsilon_1 \mid f_1(x)_i - z_i \mid}{\Delta \mathrm{fim}_Q})}{\exp(-\frac{\varepsilon_1 \mid f_1(y)_i - z_i \mid}{\Delta \mathrm{fim}_Q})}) \\
&= \prod_{i=1}^{m}(\exp(-\frac{\varepsilon_1(\mid f_1(x)_i - z_i \mid - \mid f_1(y)_i - z_i \mid)}{\Delta \mathrm{fim}_Q}) \\
&\leq \exp(\frac{\varepsilon_1 \parallel f_1(x)_i - f_1(y)_i \parallel_1}{\Delta CS_Q}) \\
&= \exp(-\frac{\varepsilon_1 \parallel f_1(x)_i - f_1(y)_i \parallel_1}{\Delta \mathrm{fim}_Q}) \\
&\leq \exp(\varepsilon_1) \ ,
\end{aligned} \tag{17}$$

where the first inequality is from triangle inequality, and the second inequality is from Equation (11). Therefore, the feature selection procedure with differentially private *F*-score satisfies $\varepsilon_1$ -differential privacy. Second, the privacy budget $\varepsilon_2$ is analyzed in the data training.

**Lemma 2.** The data training procedure satisfies $\varepsilon_2$ -differential privacy.

The neighboring datasets $D$ and $D^i$ are obtained by deleting the record $r_i$ from the dataset $D$ , and $f_2(\cdot)$ is the query of feature selection. $v_x(z)$ and $v_y(z)$ is the probability density function as following:

$$A_2(x, f_2(\cdot), \varepsilon_2) = f_2(x) + \mathrm{Lap}(\frac{\Delta CS_Q}{\varepsilon_2}) \ . \tag{18}$$

The ratio of two probability density can be represented as

$$\begin{aligned}
\frac{v_x(z)}{v_y(z)} &= \prod_{i=1}^{m}(\frac{\exp(-\frac{\varepsilon_2 \mid f_2(x)_i - z_i \mid}{\Delta CS_Q})}{\exp(-\frac{\varepsilon_2 \mid f_2(y)_i - z_i \mid}{\Delta CS_Q})}) \\
&\leq \exp(\frac{\varepsilon_2 \parallel f_2(x)_i - f_2(y)_i \parallel_1}{\Delta CS_Q}) \\
&\leq \exp(\varepsilon_2) \ ,
\end{aligned} \tag{19}$$

where first inequality is from triangle inequality, and second inequality is from Equation (4). So, the data training procedure satisfies $\varepsilon_2$ -differential privacy.

From the above analysis, the privacy budget $\varepsilon_1$ and $\varepsilon_2$ are added to the DPFSDR scheme sequentially. Combined with Theorem 2, Lemma 1 and Lemma 2, it is proved that the DPFSDR scheme satisfies ( $\varepsilon_1 + \varepsilon_2$ ) -differential privacy.

## 5 Experiments Results and Analysis

In this section, to evaluate the performance of the proposed solution, the experiments are conducted on four well-known datasets in terms of data analysis tasks and the utility for data analysis is tested with private SVM.

### 5.1 Dataset and Configuration

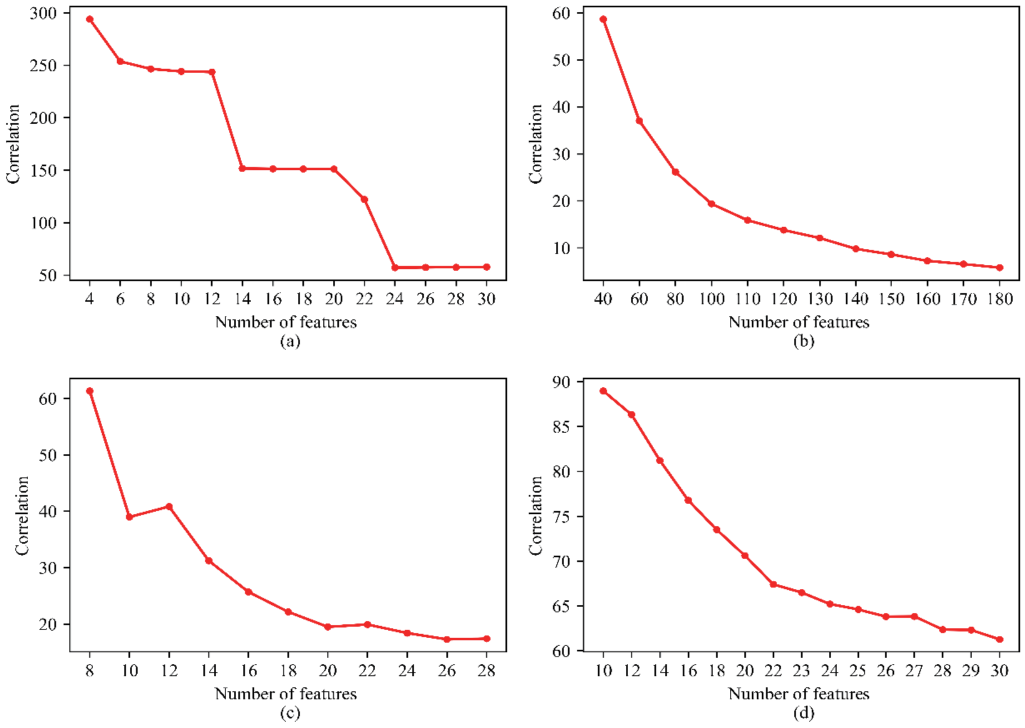
The experiments involve four datasets, WDBC, Semeion Handwritten Digit (Semeion), Dermatology, and Ionosphere, respectively. They are available in the UCI machine learning repository [26], and have different extent of data correlation and the different number of features. Each experiment is executed 1000 times. Table 1 shows the experimental datasets, given the threshold $\delta$ used to filter the attributes with the higher correlated degree, and the number of lower correlated features obtained by Algorithm 1.

**Table 1.** Descriptions of the experimental datasets, the threshold $\delta$ and No. lower correlated features

| Datasets | No. samples | No. features | No. classes | $\delta$ | No. lower correlated features |
|---|---|---|---|---|---|
| WDBC | 569 | 30 | 2 | 0.95 | 23 |
| Semeion | 1593 | 256 | 10 | 0.7 | 196 |
| Dermatology | 358 | 34 | 6 | 0.9 | 29 |
| Ionosphere | 351 | 34 | 2 | 0.95 | 33 |

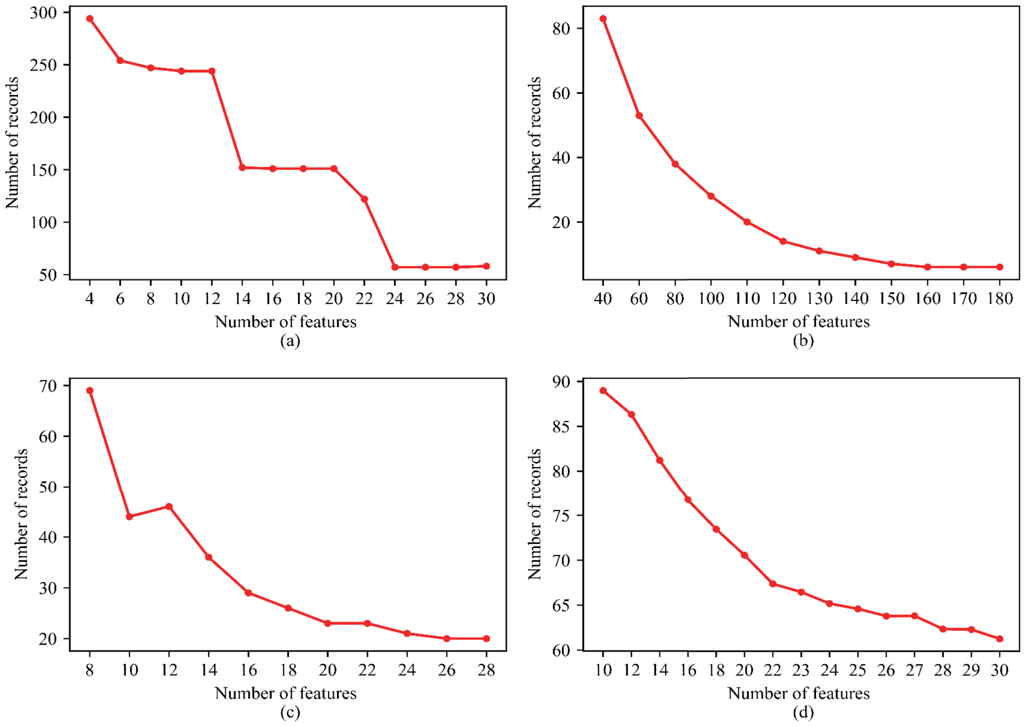
### 5.2 Data Correlation Analysis

Our proposed scheme is to improve the utility of data analysis according to the accuracy of the predicted results. SVM is chosen as a machine learning algorithm and is used to test the output perturbation to assess data utility. However, correlated data can expose more privacy information in machine learning algorithms when applying differential privacy. Not always easing to capturing the data correlation or describing accurately in the real world, previous studies do not always guarantee good performance. The proposed scheme using the weighted undirected graph reduces data correlation and can be applied to data analysis. Fig. 1 and Fig. 2 show the trend of correlation and correlated records on four different datasets with the number of features respectively.



(a) WDBC (b) Semeion (c) Dermatology (d) Ionosphere

**Fig. 1.** Correlation with the number of features on different datasets



(a) WDBC (b) Semeion (c) Dermatology (d) Ionosphere

**Fig. 2.** Number of correlated records with the number of features on different datasets

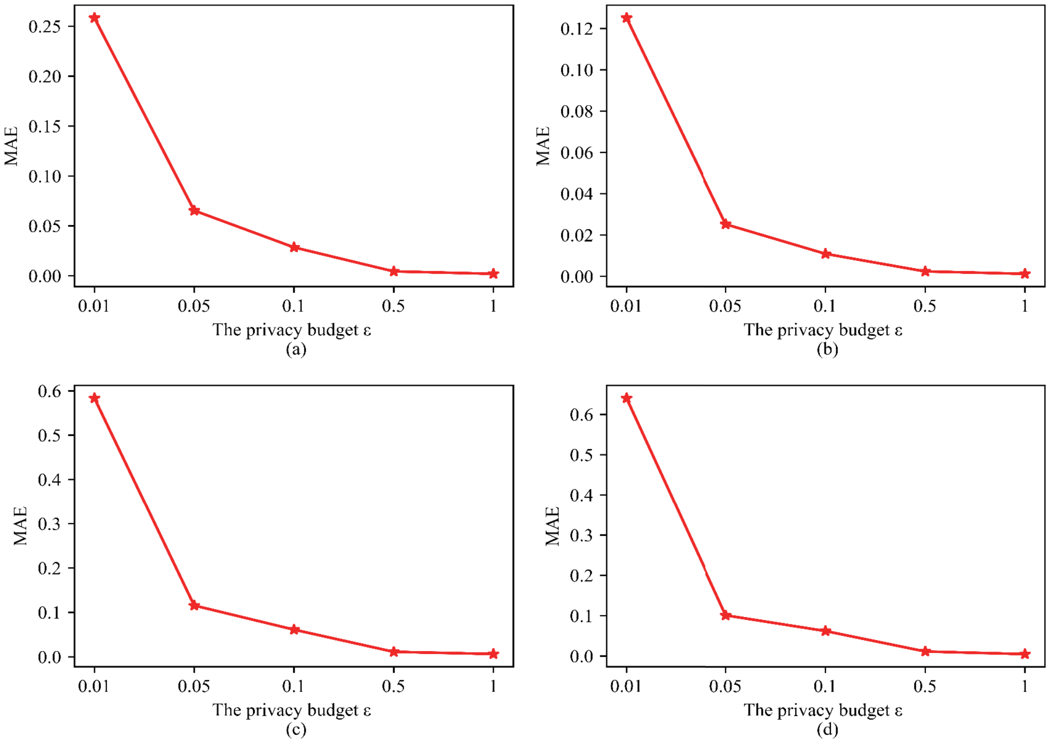
It can be seen from Fig. 1 and Fig. 2, data correlation and correlated records generally decrease with increasing number of features but eventually tend to be stable. For example, Fig. 1(a) and Fig. 1(c) demonstrate that data correlation eventually stabilizes at 24 features with the WDBC dataset and at 20 features with the Dermatology dataset. This observation in Fig. 1 indicates that data correlation across the entire dataset can be reduced while preserving a suitable number of features for data analysis because more features mean less data correlation, and the same is true for correlated records from Fig. 2. Therefore, our method reduces the data correlation effectively, and accordingly decreases the sensitivity of corrected data while maintaining good training performance.

### 5.3 Differentially Private *F*-score Feature Selection Performance Evaluation

To measure the utility of differentially private *F*-score feature selection, Mean Absolute Error (MAE) is used as performance evaluation, and set the privacy budget to $\varepsilon_1$ = 0.01, 0.05, 0.1, 0.5, 1, respectively. The MAE is defined as

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N} | \hat{f}_i(x) - f_i(x) | \ , \tag{20}$$

where $f_i(x)$ is the true statistic result for one query, $\hat{f}_i(x)$ is the perturbed statistic result, $N$ indicates the number of statistic queries. For each dataset, 1000 random statistical queries are executed, and the results are shown in Fig. 3.



(a) WDBC (b) Semeion (c) Dermatology (d) Ionosphere

**Fig. 3.** Privacy protection performance of differentially private *F*-score

As seen from Fig. 3 that the privacy protection performance changes with the privacy budgets. MAE has a downward trend as the privacy budget $\varepsilon_1$ increases, and tends to be stable toward the end, that is, the private data utility improves as the level of privacy preservation decreases. For example, Fig. 3 at $\varepsilon_1 = 0.2$ shows an MAE of around 0.028, 0.011, 0.061, 0.063 respectively for four different datasets using our proposed scheme. It can be concluded that reducing noise injecting also has a positive effect on improving the data utility.

### 5.4 Feature Selection Based on Dynamic Relevance Measure with Private SVM

To verify the effectiveness of DPFSDR on private machine learning algorithms, classification accuracy is utilized in this subsection as an indicator to assess the algorithm's performance. In the process of training data, the Laplace noise is added to the linear SVM classifier to achieve to data privacy guarantee. The feature importance sequence is calculated with $\varepsilon_1 = 0.1$, and the adding noises depend on the number of correlated records obtained similarly according to Algorithm 1, where the threshold of correlated degree between records is set to $\delta = 0.8$ and the privacy budget is set to $\varepsilon_2 = 0.01, 0.05, 0.1, 0.5, 1$, respectively. When $\varepsilon_2 = 0$, our DPFSDR algorithm is regarded as non-private-SVM, so named DFS (Dynamic Feature Selection). For better comparisons, four schemes are used in the experiments.

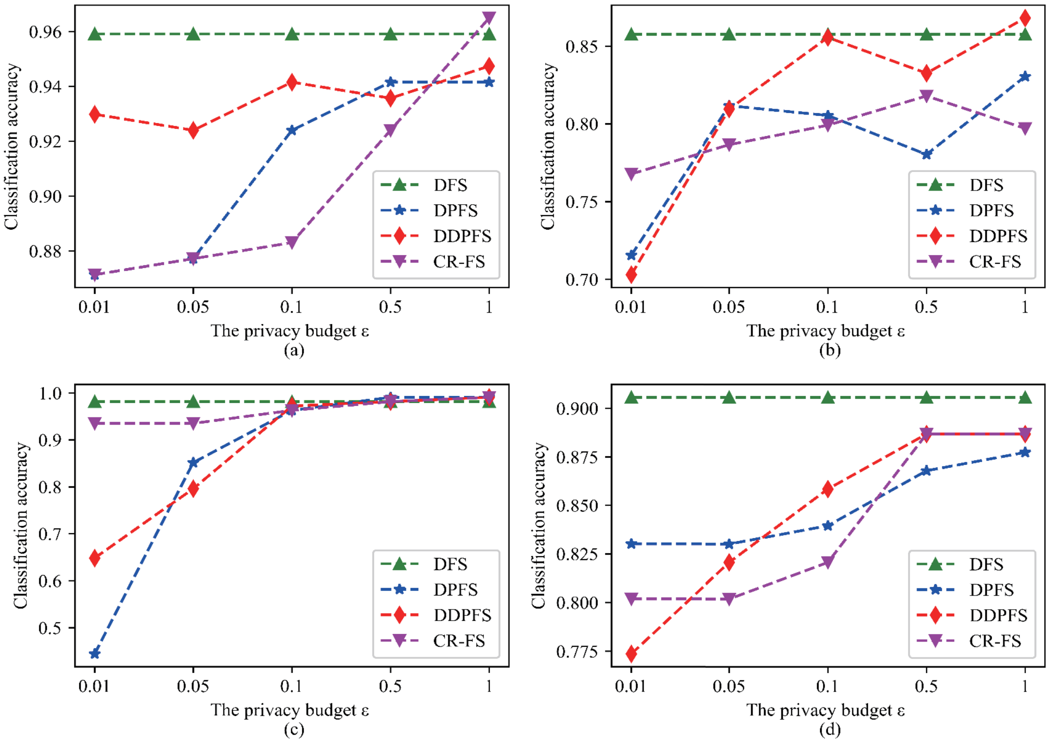
(1) DFS scheme.

(2) DPFS scheme.

(3) CR-FS scheme [22].

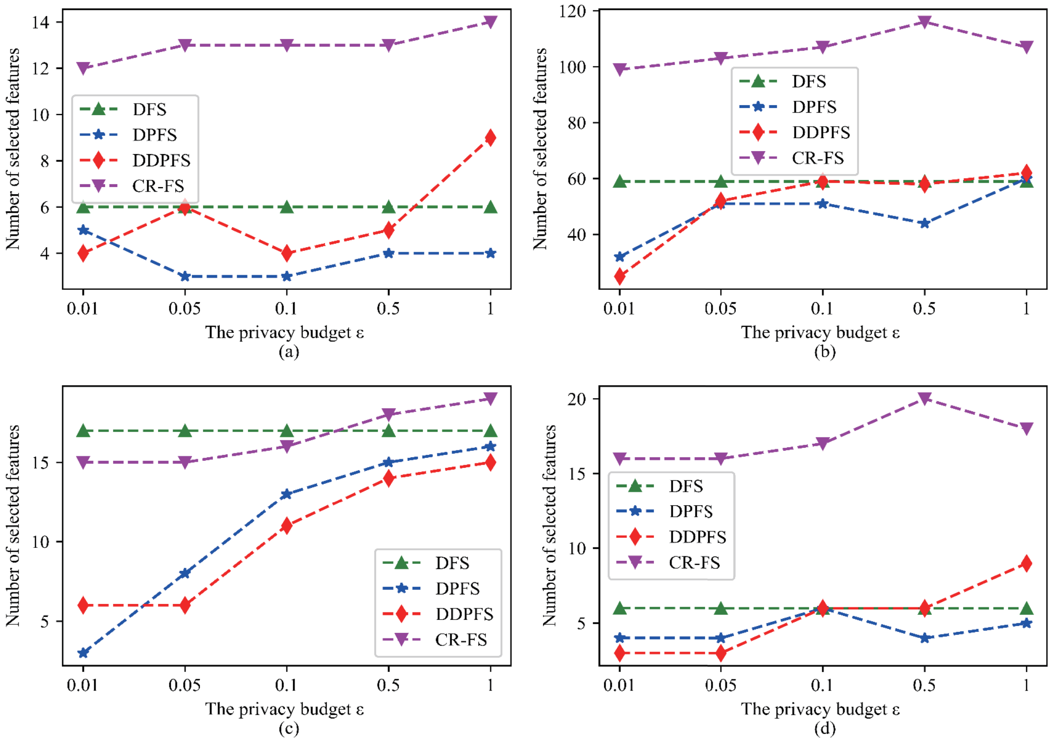
(4) Our DPFSDR scheme.

Fig. 4 demonstrates the classification accuracy varies with the growing budget $\varepsilon_2$ on four different datasets according to four schemes and corresponding the number of selected features under different budgets is shown in Fig. 5. In all cases, the classification accuracy is the average performance equal to 1000 runs of each algorithm.



(a) WDBC (b) Semeion (c) Dermatology (d) Ionosphere

**Fig. 4.** The classification accuracy under different budgets with four schemes



(a) WDBC (b) Semeion (c) Dermatology (d) Ionosphere

**Fig. 5.** The number of selected features under different privacy budgets

It can be seen from Fig. 4, that in most cases, with the increase of privacy budget, the classification accuracy of DPFS, DPFSDR, and CR-FS is gradually close to DFS. Accuracy of DFS with non-private scheme remains constant as privacy budget increases and also generally performs better than other schemes. This result shows injecting any form of privacy requirement on a dataset degrades data utility. For the three private schemes, as the privacy budget $\varepsilon_2$ increases, the privacy protection level weakens so that the accuracy tends to rise. For example, when $\varepsilon_2 < 0.1$, the classification performance of DFS is significantly better than DPFS, DPFSDR, and CR-FS; when $\varepsilon_2 \geq 0.1$, the classification accuracy of four schemes is very similar; when $\varepsilon_2 = 1$, the difference of accuracy with four schemes is less than 0.02 on WDBC and Ionosphere, and the accuracy of DPFSDR on Semeion and Dermatology is greater 0.011 and 0.002 than that of DFS respectively, that is, when $\varepsilon_2 = 1$, the classification performance with four schemes is almost the same. In general, DPFSDR is superior to DPFS and CR-FS in terms of classification performance as a whole.

The experiment in Fig. 5 exposes a rising trend of numbers of selected features with increasing privacy budget. This means the number of correlated records is reduced, and thus classification performance is improved, which is consistent with the conclusion in Fig. 4. Compared with the CR-FS, DPFS and DPFSDR select fewer features and obtain lower feature redundancy. Combined with Fig. 4 and Fig. 5, DPFSDR has better classification performance when the number of selected features is the same, and vice versa. Therefore, the DPFSDR has better utility in the feature selection process.

## 6 Conclusions

This paper focuses on dealing with the privacy leakage issue of feature selection for correlated data and proposes a differentially private feature selection based on dynamic relevance (DPFSDR) scheme, whose steps consider the data correlation and high dimension in the dataset, the accuracy of predict results, and the privacy preservation. The DPFSDR optimizes data utility by private feature selection and correlated sensitivity reduction operation. The method also provides the private machine learning algorithms to meet data analysis requirements of users and improves training accuracy effectively. According to the privacy analysis, the DPFSDR is proved to satisfy differential privacy. The DPFSDR strikes a better tradeoff between data utility and privacy leak for correlated data. The method's performance is evaluated via extensive experiments, and the results prove the DPFSDR scheme provides better utility for data analysis tasks compared to other schemes. The proposed theory and algorithm have certain theoretical and practical reference values for data analysis with correlated differential privacy.

However, the proposed private feature selection adds noise with the same amount, which may influence the data utility. In future work, adaptive differential privacy is an interesting direction, and it is used to further improve data utility by dynamic allocation of privacy budget. Other future work includes investigating high-dimensional data releasing in a distributed multi-party scenario under correlated differential privacy.

## 7 Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant No. 61762059 and No. 61862041). The authors would like to thank the helpful comments and suggestions from anonymous reviewers for this paper. The authors are grateful to the authors who provide the useful benchmark data sets.

## References

[1] W. Gao, L. Hu, P. Zhang, Class-specific mutual information variation for feature selection, Pattern Recognition 79 (2018) 328-339.

[2] C. Dwork, A. Roth, The Algorithmic Foundations of Differential Privacy, Foundations and Trends® in Theoretical Computer Science 9(3-4)(2014) 211-407.

[3] D. Kifer, A. Machanavajjhala, Pufferfish: A framework for mathematical privacy definitions, ACM Transactions on Database Systems 39(1)(2014) 1-36.

[4] D. Hemkumar, S. Ravichandra, D.V.L.N. Somayajulu, Impact of data correlation on privacy budget allocation in continuous publication of location statistics, Peer-to-Peer Networking and Applications 14(3)(2021) 1650-1665.

[5] Y. Yu, H. Zhu, M. Xie, CTP: Correlated Trajectory Publication with Differential Privacy, in: Proc. of 2021 IEEE 6th International Conference on Computer and Communication Systems (ICCCS), 2021.

[6] N. Almadhoun, E. Ayday, Ö. Ulusoy, Differential privacy under dependent tuples - the case of genomic privacy, Bioinformatics 36(6)(2019) 1696-1703.

[7] H. Wang, H. Wang, Correlated tuple data release via differential privacy, Information Sciences 560(2021) 347-369.

[8] Y. Li, J. Yang, W. Ji, Local learning-based feature weighting with privacy preservation, Neurocomputing 174(2016) 1107-1115.

[9] T.T. Le, W.K. Simmons, M. Misaki, J. Bodurka, B.C. White, J. Savitz, B.A. McKinney, Differential privacy-based evaporative cooling feature selection and classification with relief-F and random forests, Bioinformatics 33(18)(2017) 2906-2913.

[10] Z. He, A.M.V.V. Sai, Y. Huang, H. Zhang, Q. Han, Differentially private approximate aggregation based on feature selection, Journal of Combinatorial Optimization 41(2)(2021) 318-327.

[11] A. Srivastava, S. Pouyanfar, J. Allen, K. Johnston, Q. Ma, Distributed differentially private mutual information ranking and its applications, in: Proc. of 2020 IEEE 21st International Conference on Information Reuse and Integration for Data Science (IRI), 2020.

[12] Z. Liu, Y. Li, W. Ji, Differential Private Ensemble Feature Selection, in: Proc. of 2018 International Joint Conference on Neural Networks (IJCNN), 2018.

[13] X. He, A. Machanavajjhala, B. Ding, Blowfish privacy: tuning privacy-utility trade-offs using policies, in: Proc. of 2014 ACM SIGMOD International Conference on Management of Data, 2014.

[14] B. Yang, I. Sato, H. Nakagawa, Bayesian differential privacy on correlated data, in: Proc. of 2015 ACM SIGMOD International Conference on Management of Data, 2015.

[15] S. Song, Y. Wang, K. Chaudhuri, Pufferfish Privacy Mechanisms for Correlated Data, in: Proc. of 2017 ACM International Conference on Management of Data, 2017.

[16] B. Liu, T. Zhu, W. Zhou, K. Wang, H. Zhou, M. Ding, Protecting privacy-sensitive locations in trajectories with correlated positions, in: Proc. of 2019 IEEE Global Communications Conference (GLOBECOM), 2019.

[17] G. Liao, X. Chen, J. Huang, Social-aware privacy-preserving mechanism for correlated data, IEEE/ACM Transactions on Networking 28(4)(2020) 1671-1683.

[18] X. Ju, X. Zhang, W.K. Cheung, Generating synthetic graphs for large sensitive and correlated social networks, in: Proc. of 2019 IEEE 35th International Conference on Data Engineering Workshops (ICDEW), 2019.

[19] C. Liu, S. Chakraborty, P. Mittal, Dependence makes you vulnerable: differential privacy under dependent tuples, in: Proc. of 2016 Network and Distributed System Security Symposium, 2016.

[20] D. Lv, S. Zhu, Achieving correlated differential privacy of big data publication, Computers & Security 82(2019) 184-195.

[21] W. Liang, H. Chen, R. Liu, Y. Wu, C. Li, A pufferfish privacy mechanism for monitoring web browsing behavior under temporal correlations, Computers & Security 92(2020) 101754.

[22] T. Zhang, T. Zhu, P. Xiong, H. Huo, Z. Tari, W. Zhou, Correlated Differential Privacy: Feature Selection in Machine Learning, IEEE Transactions on Industrial Informatics 16(3)(2020) 2115-2124.

[23] A.L.C. Mendonça, F.T. Brito, L.S. Linhares, J.C. Machado, DiPCoDing: A differentially private approach for correlated data with clustering, in: Proc. of 21st International Database Engineering & Applications Symposium, 2017.

[24] Y. Zhang, Z. Hao, S. Wang, A differential privacy support vector machine classifier based on dual variable perturbation, IEEE Access 7(2019) 98238-98251.

[25] J. Xie, C. Wang, Using support vector machines with a novel hybrid feature selection method for diagnosis of erythemato-squamous diseases, Expert System and Applications 38(5)(2011) 5809-5815.

[26] A. Asuncion, D. Newman, UCI Machine Learning Repository. <http://www.ics.uci.edu/~mlearn/MLRepository.html>, 2007 (accessed 25.06.07).