IEICE TRANSACTIONS

on Fundamentals of Electronics, Communications and Computer Sciences

DOI:10.1587/transfun.2024TAP0011

Publicized:2024/09/09

A PUBLICATION OF THE ENGINEERING SCIENCES SOCIETY
The Institute of Electronics, Information and Communication Engineers
Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3 chome, Minato-ku, TOKYO, 105-0011 JAPAN

PAPER *Special Section on Information Theory and Its Applications*

Scalable Unified Privacy-Preserving Machine Learning Framework (SUPM)

Atsuko MIYAJI[†], **Tatsuhiro YAMATSUKI**[††], **Tomoka TAKAHASHI**[††], **Ping-Lun WANG**[†††], *and* **Tomoaki MIMOTO**[††††],

SUMMARY The widespread use of IoT devices is expected to enable the collection and utilization of a variety of data, including personal health information. For example, we could provide our personal information for machine learning operated by an external server, which in return detects signs of illness. However, it is necessary to protect privacy of personal information. Precisely, there are two issues in privacy preserving machine learning. One is data privacy which means to protect our privacy to external servers. The other is model privacy which means to protect our privacy from models. Local differential privacy (LDP) mechanisms have been proposed as a method to provide personal sensitive information to external servers with privacy protection. LDP mechanisms can ensure privacy by adding noise to data, but on the other hand, adding noise reduces their usefulness for analysis. In this paper, we propose a privacy-preserving machine learning framework, which can deal with both data privacy and model privacy. We also propose a LDP-mechanism framework which can deal with various attributes included in a single data. We also make sure feasibility of our mechanism in two cases of breast cancer screening data and ionosphere data set.

key words: *Privacy, Data availability*

1. Introduction

With the continuing development of computer science, information is being digitized, and the number of data created continues to increase. Individuals can now obtain familiar data such as their heart rate, amount of exercise conducted, and calories burned through a wearable device. It is not difficult to imagine that the speed of this trend will accelerate and that the types of data available will diversify as various IT services and the IoT expand. The collection and analysis of digital data is expected to help solve various problems. For example, in the field of health information, it is expected that real world data (RWD) and personal health care data will be used in telemedicine, health promotion, and drug development.

Despite expectations for data utilization, privacy information must be handled with care. There are various techniques used to analyze data distributed in various locations while protecting the level of privacy. Herein, we categorize them into two approaches: those that use security techniques and those that use data de-identification. The former mainly apply quasi-isomorphic cryptography [1], secret sharing [2], or garbled circuit [3] techniques to analyze data in secrecy. For example, if we assume the use of machine learning as a data analysis method, the optimization problem must be encrypted or distributed during the construction of the learning model, and the process must be iterated. Although approaches exist to deal with this bottleneck, such as approximating the activation function to speed up the process [4], the number of computations and degree of communication remain major problems. Furthermore, as the biggest problem with this approach, if the user who obtains the final output does not coincide with the data owner, it does not inherently protect the privacy of the retained data. The latter approach protects the privacy of individual data by processing the data themselves or some of the parameters. This approach can be implemented independently of the data analysis process, and it does not pose any problems in terms of the computational or communication requirements. Although various methods have been proposed for data de-identification, we focus on differential privacy in light of the increasing variety of attacks on privacy data. Differential privacy is a privacy metric, which guarantees the quantitative privacy strength based on information theory.

There are two types of differential privacy mechanisms: centralized and localized. Although both are fundamentally the same idea, we focus on LDP because the former targets the privacy of data on a large scale, while the latter is applicable in the context of individual data provision. The Laplace mechanism [5] is the most basic mechanism for satisfying differential privacy, but various task-specific methods have been proposed in terms of utility [6–9]. In recent years, a combination of federated learning and differential privacy has been proposed [10], but its practical use has not progressed much due to the difficulty of maintaining its utility and the complexity of its implementation.

In this study, we do two contribution. One is the framework of LDP-mechanisms designed for use in machine learning. In machine learning, as a single data is often composed of multiple and various attributes such as continuous and discrete data, there is a need to uniformly handle these data while adhering to privacy budgets. Therefore, in our LDP-mechanism framework for data used in ML, initially, multiple raw attributes are anonymized into discrete value data, called WA. Subsequently, WA is perturbed to a LDP data, called WALDP. Furthermore, we also give each concrete

[†] A. Miyaji is with Graduate School of Engineering, Osaka University, Japan and JAIST.

[††] T. Takahashi and T.Yamatsuki are with Graduate School of Engineering, Osaka University, Japan.

[†††] P.-L. Wang is with the Electrical and Computer Engineering department, Carnegie Mellon University, USA.

[††††] T. Mimoto is with KDDI Research, Inc., Japan.

method of WA and WALDP, which are conversion of continuous and discrete data into ordered discrete data, called *ordered-discrete anonymization (*ODA*)*, and perturbation of ODA, called ODP. ODP is a scalable unified privacy mechanism and can achieve a balance between data privacy and accuracy of the machine learning by adjusting additional parameters of dimension K, the number of classes L in ODP as well as the privacy budget ϵ. To the best of our knowledge, this study is the first to propose an LDP mechanism that can handle both continuous and discrete data in the same manner. Our proposed mechanism is not only meant for frequency analysis applications such as in [11], but also for machine learning applications.

The other is a scalable unified privacy-preserving machine learning framework, i.e., a framework that allows users to select privacy preserving methods based on the trustworthiness of the learning and testing environment. Our framework is called SUPM. SUPM consists of dimension-reduction, training, and testing. In general machine learning, dimension reduction may be performed during the learning phase. However, from the perspective of data privacy, it is necessary not only for external servers but also for data owner to conduct dimension reduction while controlling privacy. In SUPM, the dimension reduction phase and the learning phase are separated from this standpoint. This allows the realization of a scalable and uniform privacy-preserving machine learning framework for various types of data. SUPM can deal with perturbed data in the training and testing phases of machine learning methods because it functions directly on the data. This means that an institution handling a raw dataset, i.e., TTP, is not needed in both the training and testing phases. In SUPM, when the testing phase is conducted in the trusted server such as at home, it is possible to enhance performance by utilizing WA instead of perturbed data WALDP. In other words, our framework becomes customizable for data privacy based on the usage environment, allowing for a privacy-by-design approach. Such a machine learning framework is distinct from previous studies.

We also applied the proposed framework to breast cancer, ionosphere, and musk datasets in feasibility studies and confirmed the effectiveness of the framework. The results suggest that the proposed framework can generate sufficiently accurate models with the appropriate privacy strength by controlling the parameters of weak anonymization. Furthermore, we confirm that a higher accuracy is achieved despite our framework not using raw data for either the training or testing phases. Note that a preliminary version of this paper was presented at The 21th IEEE International Conference on Trust, Security, and Privacy in Computing and Communications (IEEE TrustCom 2022) [12]. The previous study applied a preliminary experiment with a relatively small data set. In this paper, we conducted experiments on a dataset with a larger number of attributes and records to confirm that our proposal will also work with more general data sets.

The remainder of this paper is structured as follows. Section 2 provides an overview of the SVMs used and introduces the LDP mechanism constituting the proposed framework. Section 3 introduces the notations used and describes the proposed framework. Section 4 presents the results of our experiments when applying the proposed framework. In section 5, we discuss our results of the experiments. Finally, section 6 provides a summary of this paper.

2. Preliminary

In this section, we first describe SVMs, the machine learning model we use, and then introduce LDP.

2.1 Support-Vector Machine (SVM)

SVMs are a type of machine learning model that solve classification and regression problems.

Assume that the training dataset has n records $\mathsf{D_i}$ $(\mathsf{i} = 1, \cdots, \mathsf{n})$, and each record $\mathsf{D_i} = [D_i, \mathsf{TA_i}] = [D_{i,1}, D_{i,2}, \cdots, D_{i,m-1}, \mathsf{TA_i}]$ has $m-1$ attributes and a target attribute $\mathsf{TA_i} \in \{-1, 1\}$. In the training phase of a linear SVM, the model calculates a function $f(D_i)$ representing a hyperplane, which is defined by an intercept b and coefficient vector $\mathbf{w} = (\mathsf{w}_1, \mathsf{w}_2, \cdots, \mathsf{w}_{m-1})$ as follows:

$$f(D_i) = \mathbf{w} \cdot D_i^T + \mathsf{b} = \sum_{j=1}^{m-1} \mathsf{w}_j \cdot D_{i,j} + \mathsf{b}.$$

Using this hyperplane, we can classify unknown data D_i' according to the output of $f(D_i')$.

$$\begin{cases} f(D_i') < 0 & \Rightarrow \mathsf{TA_i'} = -1, \\ f(D_i') \geq 0 & \Rightarrow \mathsf{TA_i'} = 1. \end{cases}$$

A limitation of linear SVM models is that they cannot correctly classify a dataset that is not linearly separable. To overcome this limitation, we use a nonlinear radial basis function (RBF) kernel, which replaces the dot product operation to a new kernel function $\exp(-\gamma||D_i - D_i'||^2)$, where γ is a nonnegative parameter [13].

In addition, when the data cannot be completely separated by a hyperplane, a soft margin can be used during the calculation of the hyperplane. This allows some training examples to be incorrectly classified, and there exists a non-negative parameter C that controls the smoothness of the hyperplane. In our experiments, we set parameter γ to $\frac{1}{(m-1)\times Var(\mathsf{D})}$, where $Var(\mathsf{D})$ is the largest variance in each dimension of D except for the target attribute, and chose parameter C such that it performs well on the raw data.

2.2 Local Differential Privacy

In the local differential privacy model [14], each of n data records has data $\mathsf{D_i}(1 \leq \mathsf{i} \leq \mathsf{n})$. Each data $\mathsf{D_i}$ contains m attributes $A_1, \cdots, A_m$. Each attribute can be discrete or continuous, the attribute has k categories $1, 2, \cdots, k$ if discrete, and has $[-1, 1]$ normalized regions if continuous. In this case, each data provider sends $f(\mathsf{D_i})$ through the random noise function f to the data collector.

Definition 1. *A function f satisfies ϵ-local differential privacy when f satisfies the following probability for all possible input data combinations x, x' and all possible output results y of f*

$$Pr[f(x) = y] \le exp(\epsilon) \cdot Pr[f(x') = y].$$

The Piecewise mechanism [11], denoted by PW in this paper, is a random noise function to satisfy LDP for continuous values, which is shown in Algorithm 1. The probability density function $\mathrm{pdf}(y|x)$ that the output of PW follows is

$$\mathrm{pdf}(y|x) = \begin{cases} p, & \text{if} \quad x \in [l, r], \\ \frac{p}{exp(\epsilon)}, & \text{if} \quad x \in [-H, l) \cup (r, H], \end{cases}$$

where $p = \frac{\exp(\epsilon) - \exp(\epsilon/2)}{2\exp(\epsilon/2)+2}, H = \frac{exp(\epsilon/2)+1}{exp(\epsilon/2)-1}, l = \frac{H+1}{2} \cdot x_j - \frac{H-1}{2}$, and $r = l + H - 1$.

Algorithm 1 Piecewise mechanism (PW) [11]

```
Require: continuous value $x_j$, range $[-t, t]$, privacy budget $\epsilon$
Ensure: perturbed data $y_j$
Adapt range $[-t, t]$ to $[-1, 1]$
Sample $R$ uniformly at random from $[0, 1]$
Compute $H = \frac{exp(\epsilon/2)+1}{exp(\epsilon/2)-1}, l = \frac{H+1}{2} \cdot x_j - \frac{H-1}{2}, r = l + H - 1$
if $R \le \frac{e^{\frac{\epsilon}{2}}}{e^{\frac{\epsilon}{2}}+1}$ then
    Sample $y_j$ uniformly at random from $[l, r]$
else
    Sample $y_j$ uniformly at random from $[-H, l) \cup (r, H]$
end if
return $t \cdot y_j$
```

The Randomised Response mechanism [15], denoted by RR in this paper, is a random noise function to satisfy the LDP for discrete values. The input x and the output y have L types as well. RR adds noise in the following way

$$p(y|x) = \begin{cases} \frac{exp(\epsilon)}{L-1+exp(\epsilon)}, & \text{if} \quad y = x, \\ \frac{1}{L-1+exp(\epsilon)}, & \text{if} \quad y \neq x. \end{cases}$$

RR outputs the same or a different value with the probability of $\frac{exp(\epsilon)}{L-1+exp(\epsilon)}$ or $\frac{1}{L-1+exp(\epsilon)}$, respectively. RR is given in Algorithm 2.

Algorithm 2 Randomised Response mechanism (RR) [15]

```
Require: discrete value $x_j$ of $A_j$, $L$ values $\{A_j[1], \cdots, A_j[L]\}$, privacy budget $\epsilon$
Ensure: perturbed data $y_j$
Sample $x$ uniformly at random from $[0, 1]$
if $x \le \frac{exp(\epsilon)}{L-1+exp(\epsilon)}$ then
    $y_j = x_j$
else
    Sample $y_j$ uniformly at random from$\{A_j[1], \cdots, A_j[L]\}$ except $x_j$
end if
return $y_j$
```

3. Proposed Privacy-Preserving Machine Learning Framework

Machine learning generally consists of two phases: training and testing. The training phase constructs the model, whereas the testing phase uses data to test the model. In general, the model-building and operation servers are assumed to be trustworthy and are therefore called trusted third parties (TTPs). However, the risk of a data leakage through a cyber-attack cannot be reduced to zero. Constructing an absolutely secure TTP is difficult; therefore realizing substantial privacy protection is also difficult concerning data utilization based on TTPs.

In this paper, we propose a privacy mechanism and privacy-preserved framework for machine learning. Our framework protects against privacy leakage in the constructed model as well as the server that constructs or operates the model to test the data. The proposed framework does not assume TTPs exist in either the training or testing phases; thus, each data owner transmits their own data. Our comprehensive privacy-enhanced framework is a novel approach.

3.1 Main Concepts

Before explaining our concept, we present the various notation used in our paper.

- ML: machine learning
- LDP: local differential privacy
- PW: piecewise mechanism [11] and data perturbed by PW is called PW data
- RR: randomized response mechanism
- WA: weakly anonymized data
- WALDP: perturbation of WA
- ODA: proposed ordered-discrete anonymization and data anonymized by ODA is called ODA data.
- ODP: proposed ordered-discrete perturbation and data perturbed by ODP is called ODP data.
- SUPM: proposed privacy-preserving machine learning framework
- SUPM.ext: proposed extended privacy-preserving machine learning
- PPTraining: proposed privacy-preserving training
- PPTesting: proposed privacy-preserving testing
- Agg: aggregator
- ϵ: privacy budget
- ϵ_K: privacy budget ϵ/K
- n: total number of records
- m: total number of attributes of one record (called dimension)
- K: number of attribute used from m attributes
- TA: target attribute
- A_j:j-th (continuous/discrete) attribute ($j \in [1, m-1]$) (excluding TA)
- D, $\mathsf{D_i}$: record, the i-th record, $i = 1, \cdots, n$, which consists of m data of m attributes, $\mathsf{D_i} = [D_i, \mathsf{TA}_i] = [D_{i,1}, \cdots, D_{i,m-1}, \mathsf{TA}_i]$
- $D_{i,j}$: the j-th attribute of the i-th record $\mathsf{D_i}$ ($j \in [1, m-1]$).
- $\mathsf{TA_i}$ is a target attribute of $\mathsf{D_i}$.
- $\max(A_j), \min(A_j)$: the maximum or minimum value

of attribute A_j (We can use these notations in both continuous- and discrete attribute transforming discrete data to ordered-discrete data, which is shown below.)

- $\mathsf{Range}_j = \max(A_j) - \min(A_j)$: range of attribute A_j
- L: number of setting classes of attribute
- $A_{j_1}, \cdots, A_{j_K}$: chosen attributes
- $\mathsf{OD}_j[1], \cdots, \mathsf{OD}_j[L]$: ODA-anonymized data of attribute A_j

To construct the machine learning model, several existing protocols assume the existence of a trusted party or client who collects or handles a set of raw data [10, 16]. However, our privacy-enhanced framework does not assume that a TTP collecting or handling raw data exists. Each data owner manages their own data. This is a major difference between the existing studies and ours. In general, privacy protection and machine learning performance have conflicting characteristics. LDP mechanisms are powerful in terms of data privacy protection, but they degrade the machine learning performance. Therefore, existing protocols use DP mechanisms to influence the trained parameters of local (trusted) clients [10] or TTPs [16]. Here, data with LDP are called LDP data. A balance between privacy and the machine learning performance is difficult to achieve when using only raw or LDP data.

To construct a new privacy mechanism fit for machine learning without assuming TTPs, we focus on the data characteristics, i.e., a dataset that consists of multiple attributes. The characteristics of the attributes are diverse, with some continuous and discrete values, as seen in [17, 18]. Furthermore, LDP mechanisms for continuous [11] and discrete data [19] have typically been independently constructed. A single mechanism that can handle both types of data in a unified manner and achieve a proper balance would be scalable. We define a new notion of a unified data, *ordered-discrete* data, to solve the problem of data diversity. Then we propose a method for annonymizing both continuous and discrete data to called L-ordered-discrete data in a unified manner, which is denoted by ODA. ODA data is a weakly anonymized data and take an intermediate position between raw and LDP data. Finally, we propose a privacy mechanism that first converts any raw data into ODA and then perturbs ODA into LDP data, which is called ordered-discrete perturbation (ODP).

Next, we investigate how to control the privacy and accuracy of machine learning, where perturbed data are used for both training and testing phases. A record D_i consists of multiple attributes $\{A_j\}$. From a privacy perspective, if the privacy budget for each attribute is ϵ, and the number of attributes is m, then the total privacy budget for a single record becomes $m\epsilon$. More privacy is wasted as the number of attributes increases. We refer to the number of attributes in a record as *the dimension*. In PW [11], the number of used attributes, K, was determined according to the privacy budget, and a data owner randomly selected K attributes from all m attributes, perturbed the K attributes, set the remaining $m - K$ attributes to zero, and sent all m attributes to Agg. Although their method works well for operations such as averaging, achieving good accuracy with machine learning remains difficult. Because a value of zero is ignored for averaging but it is meaningful for machine learning. In addition, they do not propose an integrated way to handle both the training and testing phases of machine learning. Therefore, if each data owner randomly selects K attributes and perturbs the selected data randomly in testing phase, then it will only degrade the accuracy of machine learning.

In a general machine learning, the dimension reduction is performed during the learning phase for performance reasons. This is done within the learning server. However, this method exposes the user's privacy to the server at the time of dimension reduction. Therefore, we separate the dimension reduction phase from the learning and test servers, i.e., divide machine learning into three phases, dimension reduction, learning, and test, and realize privacy-preserving machine learning that considers the user's privacy in each phase. In other words, we propose a framework for dimension reduction while considering the user's privacy, rather than dimension reduction for the sole purpose of accuracy, as has been conventionally done in machine learning.

3.2 Privacy Mechanism Framework for Machine Learning

To build a framework for LDP-mechanisms with the purpose of utilization in ML, the necessary factors are described. In ML, as a single data is often composed of multiple and various attributes, it is essential to handle continuous and discrete data uniformly while adhering to privacy budgets. The main concept involves the transformation of raw data from multiple and various attributes into weakly anonymized data (WA). Subsequently, this is further transformed into perturbed data (WALDP). The key point is to achieve this while maintaining control over the privacy budget.

First, let's explain the basic concept of weak anonymization for multiple and various attributes. Consider multiple attributes initially. For instance, let's assume we have data for each city, with monthly temperature $A[i]$, Consumer Price Index (CPI) $B[i]$, and estimated population $C[i]$ for each month ($i = 1, \cdots, 12$). Then, the total number of attributes for a city is 36. If we add perturbation noise ϵ to all raw data directly, it would result in wasting 36ϵ for the data of one city. This is the perturbation method based on raw data. Instead of using raw data directly, the idea of WA involves anonymization of raw data while suppressing privacy budget and not degrading the utility of ML. For example, in this case, rather than using monthly data, we could transform it into the average of every three months. As a result, the number of attributes is reduced from 36 to 12, and the privacy budget waste is reduced to 12ϵ. This is a concept of WA-data from raw data.

Next, we will discuss the handling of various attributes, namely, continuous and discrete data. Discrete data can be divided into ordered and unordered discrete data. When perturbing based on the type of attribute, i.e., continuous, ordered or unordered discrete data, multiple types of perturbations are applied to one single data. This is the perturbation method based on raw data. Instead of using raw data directly, the idea of WA involves transforming various raw data into ordered discrete data, suppressing privacy budget, and not degrading the utility of ML. In this way, raw data with multiple and various attributes can be uniformly transformed into WA data. The next phase involves applying perturbation methods to WA data to construct LDP-data (WALDP). This constitutes the framework of LDP mechanisms for data used in ML.

3.3 Instantiation of Privacy Mechanism for Machine Learning

In this subsection, we propose our privacy mechanism called the ordered-discrete-pertubation mechanism (ODP-mechanism), which is a unified privacy mechanism that can be used for any data type. ODP consists of three transformation functions. The first is a discrete-to-ordered-discrete transformation (DTO). The second function transforms any data into L-ordered-discrete data, called L-ordered-discrete data anonymization, ODA. With ODA, raw data are weakly anonymized. Finally, the L-ordered-discrete data are perturbed through a third function. Our privacy mechanism therefore can handle data in a uniform manner, regardless of whether the data are continuous or discrete. Notably, the data owner can execute this mechanism.

First, we describe DTO. Let A_j be a discrete attribute. Discrete attributes differ from continuous attributes in that large and small comparisons can be difficult. For example, consider a discrete attribute A_j of *directions*, which consists of four datums of north, east, south, and west. Directions are not directly comparable. Therefore, to compare the discrete data, we formally assign numeric labels $i = 1, 2, \cdots$ to the discrete data, which are called *ordered-discrete* data. The discrete attributes can then be ordered based on their label. Thus, first- or last-label discrete data can be regarded as the minimum or maximum discrete data, denoted as $\min(A_j)$ and $\max(A_j)$, respectively. We can then use the notation $\mathsf{Range}_j = \max(A_j) - \min(A_j)$ for discrete data. Here, the number of classes of discrete attribute A_j equals $\mathsf{Range}_j + 1$. That is, the same notation of $\min(A_j)$, $\max(A_j)$, or Range_j for continuous data can be used for discrete data. In the directions example, let $A_j[1]$ =north, $A_j[2]$ =east, $A_j[3]$ =south, and $A_j[4]$ =west, and thus $\min(A_j) = 1$, $\max(A_j) = 4$, and $\mathsf{Range}_j = 3$. DTO is then given inputs of attribute east and {north, east, south, and west} and outputs $(2, \{1, 2, 3, 4\})$, as indicated in Algorithm 3. Note that DTO is initially determined and lets data owners know.

Algorithm 3 Discrete-to-ordered-discrete data (DTO)

```
Require: data $x_j$ of discrete attribute $A_j$
Ensure: order index $ox_j$ of $x_j$ and indices of ordered-discrete $A_j$
    $\{\min(A_j), \cdots, \mathsf{Range}_j+1\}$
```

Then, the subsequent transformation anonymizes both order-discrete and continuous data in Algorithm 4. Algorithm 4 is called *ordered-discrete*-anonymization, ODA. In Algorithm 4, both continuous and discrete data are anonymized to L-ordered-discrete data, ODA.

Algorithm 4 Ordered-Discrete Anonymization (ODA)

```
Require: data $x_j$ of attribute $A_j$, $\min(A_j)$, $\mathsf{Range}_j$, number of classes $L$
Ensure: weakly anonymized data $y_j$ and $L$ weakly anonymized classes of
    $A_j$ $\{\mathsf{OD}_j[i]\}$
if $A_j$ is continuous data then
    $\mathsf{OD}_j[1] \leftarrow \min(A_j) + \mathsf{Range}_j/2L$
    for $i = 2$ to $L$ do
        $\mathsf{OD}_j[i] \leftarrow \mathsf{OD}_j[i-1] + \mathsf{Range}_j/L$
    end for
    $i \leftarrow \lceil \frac{(x_j - \min(A_j))L}{\mathsf{Range}_j} \rceil$
    $y_j \leftarrow \min(A_j) + (2i-1)\mathsf{Range}_j/2L$
else
    $(ox_j, \{\min(A_j), \cdots, \mathsf{Range}_j + 1\}) \leftarrow \mathsf{DTO}(\mathsf{x_j}, \mathsf{A_j})$.
    if $\mathsf{Range}_j \le L$ then
        $\ell \leftarrow \mathsf{Range}_j + 1$
    else
        $\ell \leftarrow L$
    end if
    for $i = 1$ to $\ell$ do
        $\mathsf{OD}_j[i] \leftarrow j_i$
    end for
    $i' \leftarrow ox_j \pmod{L}$,
    $y_j \leftarrow \mathsf{OD}_j[i']$,
end if
return $y_j$ and $\{\mathsf{OD}_j[i]\}$
```

Finally, all L-ordered-discrete data are perturbed in Algorithm5, which is called *ordered-discrete perturbation*, ODP. Algorithm 5 can handle both continuous and discrete data in a unified manner through ODA. Algorithm5 calls DTO(Algorithm 3) for discrete data, and ODA(Algorithm 4) and RR(Algorithm 2). In the existing mechanisms [11, 19], only ϵ is used to control the privacy and accuracy of the machine learning, which is a trade-off. Our mechanism controls the privacy and accuracy of machine learning using the parameters of privacy budget ϵ and ODA's parameter L, thereby enabling smoother control.

Algorithm 5 Ordered-Discrete Perturbation (ODP)

```
Require: data $x_j$ of attribute $A_j$, $\min(A_j)$, $\mathsf{Range}_j$, number of classes $L$,
    privacy budget $\epsilon$
Ensure: data $z_j$
if $A_j$ is discrete data then
    Transform them to ordered-discrete data by DTO,
    $(x_j, \{\min(A_j), \cdots, \mathsf{Range}_j + 1\}) \leftarrow \mathsf{DTO}(x_j, A_j)$.
end if
Annonymize them by ODA,
$(y_j, \mathsf{OD}_j[1], \cdots, \mathsf{OD}_j[L]) \leftarrow \mathsf{ODA}(x_j, \min(A_j), \mathsf{Range}_j, L)$.
Perturb them by RR
$z_j \leftarrow \mathsf{RR}(y_j, \mathsf{OD}_j[1], \cdots, \mathsf{OD}_j[L], \epsilon)$.
return $z_j$
```

3.4 Privacy-Preserving Machine Learning Framework SUPM

Here, we propose a privacy-preserving machine learning framework (SUPM). SUPM consists of three major phases: dimension-reduction, training, and testing. The dimension-reduction phase reduces the number of attributes because large number of attributes waste the privacy budget. Although WA can also reduce the dimension described in Section 3.2, in the dimension reduction phase, we focus on a method to reduce the dimension by interacting between data owner and Agg while maintaining privacy. In some datasets, attributes that indicate good performance for machine learning are known. In such a case, training and testing are applied on the attributes with a good performance without executing the dimension-reduction phase.

Herein, we propose three methods for selecting the attributes while protecting the data privacy. One method involves perturbing the data with PW, called DR.PW. The other methods involve applying the proposed ODA or ODP, called DR.WA or DR.WALDP, respectively. Since the only difference between DR.WA and DR.WALDP is whether the subroutine is ODA or ODP, only DR.WA is explained in detailed.

Algorithm 6 shows DR.PW. In DR.PW, for given number of used attributes, K, each data owner chooses K attributes, perturbs them using PW, and then sends them to Agg. Then Agg determines K attributes by applying data sent by owners.

Algorithm 6 Dimension reduction with PW (DR.PW)

```
Require: $m$-dimension raw data $\mathsf{D_i}$= $[D_{i,1},\cdots,D_{i,m-1},\mathsf{TA_i}]$, privacy
    budget $\epsilon$, the number of used attribute $K$
Ensure: chosen attributes $A_{j_1},\cdots,A_{j_K}$
  $\epsilon_{K+1} \leftarrow \epsilon/(K+1)$
  Sample $K$ data of $(D_{i,j_1},\cdots,D_{i,j_K})$ and target attribute $\mathsf{TA}_i$ from $m$-
    dimension data $\{\mathsf{D_i}\}$ uniformly, execute $\mathsf{PW}(\mathsf{x_{i,j_t}}, \mathsf{Range}, \epsilon_{\mathsf{K+1}}),\cdots,$
    $\mathsf{PW}(\mathsf{x_{i,j_K}}, \epsilon_{\mathsf{K+1}}), \mathsf{PW}(\mathsf{TA_i}, \epsilon_{\mathsf{K+1}})$, and send them to Agg.
  Agg collects $\{(\mathsf{PW}\ (D_{i,j_1},\epsilon_{K+1}),\cdots,\mathsf{PW}\ (D_{i,j_K},\epsilon_{K+1}),\mathsf{PW}$
    $(\mathsf{TA}_i,\epsilon_{K+1})\}$, determines $K$-attribute $A_{j_1},\cdots,A_{j_K}$ by evaluating
    these correlation coefficients without seeing any raw data.
  return $K$-attribute $A_{j_1},\cdots,A_{j_K}$
```

Algorithm 7 shows DR.WA. Because we focus on the use of SVMs in Sections 2 and 4, a binary classification is applied in DR.WA. For simplicity, let $\mathsf{TA} = \{-1, 1\}$. In DR.WA, for the given number of attributes used, K, each data owner chooses K attributes, anonymizes them through ODA, computes the correlation coefficients by applying the annoymized attributes, and sends them to Agg. Then, Agg determines K attributes by applying the data sent by the owners. Algorithm 7 describes dimension reduction using ODA, but if ODP is used instead of ODA, it becomes a dimension reduction with perturbations added to the data, which is called DR.WALDP. Since experiments in DR.PW was conducted in this paper as a method with perturbations added, DR.WALDP is omitted.

Algorithm 7 Dimension reduction with ODA (DR.WA)

```
Require: $m$-dimension raw data D= $[D_{i,1},\cdots,D_{i,m-1},\mathsf{TA_i}]$,
    $\min(A_j)$, $\mathsf{Range}_j$, the number of setting classes of attribute $L$,
    the number of used attribute $K$
Ensure: chosen attributes $A_{j_1},\cdots,A_{j_K}$
  Sample $K$ data of $(D_{i,j_1},\cdots,D_{i,j_K})$ and target attribute $\mathsf{TA}_i \in$
    $\{-1,1\}$ uniformly, get $\{y_{j_s}\}$ by executing $\{\mathsf{ODA}(D_{i,j_s},\min(A_{j_s}),$
    $\mathsf{Range}_{j_s}, L)\}$, compute $y_{j_s}\cdot\mathsf{TA}_i$ for $s=1,\cdots,K$, and send $K$-tuple
    data to Agg.
  Agg collects perturbed parts of correlation coefficients and determines
    $K$-attribute $A_{j_1},\cdots,A_{j_K}$ without seeing any raw data.
  return $K$-attribute $A_{j_1},\cdots,A_{j_K}$
```

In this study, we also conducted experiments on randomly reducing the number of attributes up to a defined number for comparison with our DR.PW and DR.WA. The method of random attribute reduction is called DR.Rand.

Next, we describe the training and testing phases. In the dimension-reduction phase, the attributes to be used in training and testing are determined; thus, using the determined attributes, privacy-preserving training and testing are executed, which are denoted as PPTraining and PPTesting, respectively. Here, PPTraining and PPTesting are described formally in Algorithms 8 and 9. In PPTraining, the data owner perturbs raw data x of the determined attributes using ODP and sends the perturbed data to Agg, where Agg conducts the training and builds the model using the perturbed data. In PPTesting, the data owner perturbs raw data x of the determined attributes using ODP and sends the perturbed data to the training model, where the training model executes the perturbed data and returns a result.

Note that we can extend PPTraining and PPTesting by using four combinations of data types of PPTraining and PPTesting, which are (training, testing)= (ODP, ODP), (ODP, ODA), (ODA, ODP), and (ODA, ODA). In addition, these combinations are combined with three types of dimension-reduction of DR.WA, DR.PW, and DR.Rand. In summary, SUPM consists of three types of dimension reduction, PPTraining (Algorithm 8), and PPTesting (Algorithm 9); and SUPM.ext extends PPTraining and PPTesting to four combinations of (ODP, ODP), (ODP, ODA), (ODA, ODP), and (ODA, ODA). To compare our SUPM with PW, we examine a case in which the raw data are perturbed through PW and (training, testing)= (PW, PW).

Algorithm 8 Privacy-Preserving Training (PPTraining)

```
Require: $K$ data and target data, $[D_{i,j_1},\cdots,D_{i,j_K},\mathsf{TA_i}]$, the num-
    ber of setting classes of attribute $L$, the privacy budget $\epsilon$
Ensure: trained model
  $\epsilon_{K+1} \leftarrow \epsilon/(K+1)$
  Sample $K$ data of $(x_{i,j_1},\cdots,x_{i,j_K})$ and target data $\mathsf{TA}_i$.
  $y_{j_s} \leftarrow \mathsf{ODP}(x_{i,j},\min(A_j),\mathsf{Range}_j,\epsilon_{K+1})$ for $j=1,\cdots,K$.
  $y_{j_{K+1}} \leftarrow \mathsf{ODP}(\mathsf{TA}_i,-1,2,\epsilon_{K+1})$.
  Send $K+1$-tuple perturbed data to Agg.
  Agg collects perturbed $K+1$ data and constructs training model.
  return Training model.
```

Algorithm 9 Privacy-Preserving Testing (PPTesting)

```
Require: $K$ data and target data $[D_{i,j_1},\cdots,D_{i,j_K},\mathsf{TA_i}]$, the num-
    ber of setting classes of attribute $L$, the privacy budget $\epsilon$
Ensure: Result.
  $\epsilon_{K+1} \leftarrow \epsilon/(K+1)$
  Sample $K$ data of $(x_{i,j_1},\cdots,x_{i,j_K})$ and target data $\mathsf{TA}_i$.
  $y_s \leftarrow \mathsf{ODP}(x_{i,s},\min(A_s),\mathsf{Range_s},\epsilon_{K+1})$ for $s=j_1,\cdots,j_K$.
  $y_{j_{K+1}} \leftarrow \mathsf{ODP}(\mathsf{TA}_i,-1,2,\epsilon_{K+1})$,
  Send $K+1$-tuple perturbed data to a training model.
  A training model executes perturbed $K+1$ attributes and obtains the
    result.
  return Result.
```

Theorem 1. SUPM *that applies* DR.PW *for dimension-reduction satisfies local differential privacy.*

Proof. Let $\mathsf{D} = \mathsf{D_{train}} \cup \mathsf{D_{test}}$ be a dataset to be privacy preserved. SUPM consists of a dimension-reduction process and an attribute perturbation process. In the dimension-reduction process, SUPM applies DR.PW. SUPM applies

ODP in the attribute perturbation process. ODP satisfies LDP because it gives outputs through RR, which is proved to satisfy LDP [15]. Since these processes are applied to $\mathsf{D}_{\mathsf{train}}$ in PPTraining and to $\mathsf{D}_{\mathsf{test}}$ in PPTesting, SUPM satisfies LDP for D. □

Note that DR.WA and ODA does not satisfy LDP. However, if we use DR.WALDP instead of DR.WA, then SUPM with DR.WALDP also satisfies LDP in the same way as Theorem 1. Our goal is to provide a framework that works in various ML environments. In our framework, ML consists of three phases of dimension reduction, learning, and testing separately. In other words, each method can be selected based on whether the environment for dimension reduction, learning, or testing is reliable or unreliable. That is, for example, if the environment for (dimension reduction, learning, testing) is (reliable, unreliable, reliable), then each choice is (DR.WA, ODP, ODA).

4. Feasibility Studies

4.1 Experimental Remarks

In this study, three datasets were obtained from the UCI machine learning repository, i.e., the Breast Cancer Wisconsin (Diagnostic) dataset (WDBC) [17], the Ionosphere dataset [18], and the Musk dataset [20], and were used to evaluate the feasibility of our mechanism. The WDBC and Ionosphere datasets are sized with less than 1000 instances and approximately 30 attributes. We also conductd experiments using the Musk dataset, which is a large datasets, to see if differences occur from discrepancies differences in the number of instances and attributes.

These datasets were selected because they concern binary classification tasks, which are particularly suitable for SVMs. Note that our proposed mechanism is still applicable to any machine learning dataset and is not limited to binary classification tasks. To evaluate the performance of our mechanism, we used the following four types of data to train SVM models and compare their accuracies.

1. **Raw data**: We use the original raw data. This experiment indicates the maximum accuracy achievable for our mechanism, which is executed in Experiment 2.
2. PW **data**: We apply PW to the data. This experiment is used as the comparison with our mechanism, which is executed in Experiment 2.
3. SUPM **data**: We apply SUPM to anonymize the data. Experiment 1 was conducted for all possible combinations with $K = [2, 10]$ and $L = [2, 5]$ for the SUPM data, and two or three good combinations were reported. Experiment 2 reports the best accuracy for each privacy budget among the good (K, L) combinations found in Experiment 1.
4. SUPM.ext **data**: In SUPM.ext, there are four combinations of (PPTraining, PPTesting)= (ODP, ODP), (ODA, ODP), (ODP, ODA), and (ODA, ODA). Experiment 3 tests these combinations and checks their performances.

The above three experiments are described below.

- **Experiment** 1: Experiment combinations of (K, L) in the SUPM data, to find the optimal (K, L). We report two or three combinations providing good results in DR.Rand.
- **Experiment** 2: Experiment Raw data, PW data, and SUPM data for each case. The dimension reduction phase of SUPM data is performed for DR.PW, DR.Rand, and DR.WA, respectively.
- **Experiment** 3: Experiment four combinations of (PPTraining, PPTesting) in SUPM.ext.

For normalized data, we split the dataset into ten parts and conducted cross-validation to measure the accuracy of our model. Note that we randomly shuffled the dataset prior to the ten-fold cross-validation process. We used the same random seed for splitting, dimension reduction, and perturbation for the three types of data, to ensure that we conducted the same training and testing processes.

Our experiments were conducted on a Ubuntu 20.04 machine with an Intel Xeon Gold 5120 CPU and 48 GB of RAM. We constructed our SVM models using Python 3.8 and scikit-learn [13], which is a machine learning library.

4.2 WDBC Dataset

The WDBC dataset consists of 569 instances diagnosing breast cancer as benign or malignant. Thirty continuous attributes are computed from a digitized image. In the WDBC dataset experiments, we used $C = 2.1$ as the SVM parameter.

4.2.1 Raw data

Figure 2 shows that when we directly use the WDBC dataset for training and testing, we can achieve an accuracy of 98.04%. Therefore, the maximum accuracy we can achieve on this dataset is 98.04%, and we compared this with other experimental results.

4.2.2 PW data

When the privacy budget ϵ is smaller than 50, the accuracy of PW data is at most 84.77%, which is lower than that of the SUPM data.

4.2.3 SUPM data

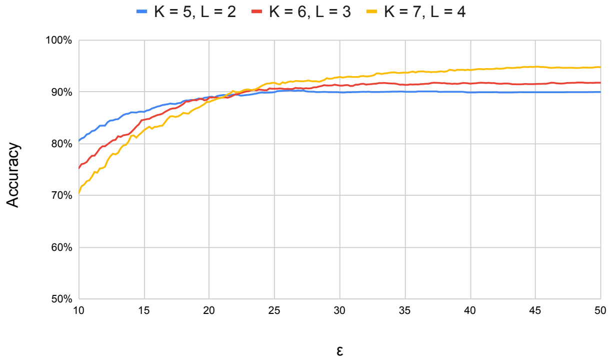


Fig. 1 Accuracy when different values of K and L are configured in the WDBC dataset. The x-axis is the privacy budget we use for training and testing, and the y-axis is the accuracy. SUPM peturbed in both the training and testing phases, and DR.Rand was applied for a dimension reduction.

Figure 1 shows the results of three different configurations of $(K, L) = \{(5, 2), (6, 3), (7, 4)\}$ by SUPM with DR.Rand. The greater the number of attributes or classes, the stronger noise effect that occurred. Therefore, we observed a more drastic change in accuracy for $(K, L) = (6, 3)$ than for $(K, L) = (5, 2)$, owing to the increase in the privacy budget ϵ. Similar trends were observed for $(K, L) = (6, 3)$ and $(K, L) = (7, 4)$. For $(K, L) = (5, 2)$, no significant change in accuracy was observed to have resulted from the privacy budget. That is, even for a tight privacy budget, few attributes and classes achieve a high accuracy. Moreover, for $(K, L) = (7, 4)$, the highest accuracy tends to be achieved with a privacy budget $\epsilon \geq 21.4$. A similar trend was observed for DR.PW and DR.WA.

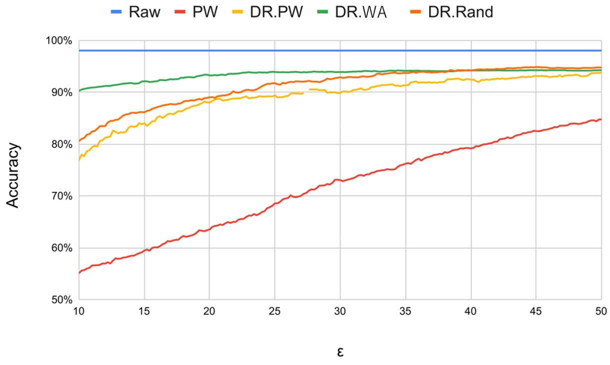


Fig. 2 Experimental comparison of raw data, PW, and SUPM-DR.PW, DR.WA and DR.Rand data in the WDBC dataset. The x-axis is the privacy budget we applied during the training and testing phases, and the y-axis is the accuracy.

Figure 2 shows the highest accuracy listed below.

- DR.PW : $(K, L) \in \{(2, 2), (4, 4)\}$
- DR.WA : $(K, L) \in \{(2, 2), (4, 4)\}$
- DR.Rand : $(K, L) \in \{(5, 2), (7, 4)\}$

For the DR.WA mechanism, attributes can be selected with a higher accuracy than that of DR.PW or DR.Rand. Even with the smallest privacy budget $\epsilon = 10$ used in this experiment, the accuracy exceeds 90.29% for DR.WA. The accuracy exceeds 90% when $\epsilon = 27.4$ and 22.4 for DR.PW and DR.Rand, respectively.

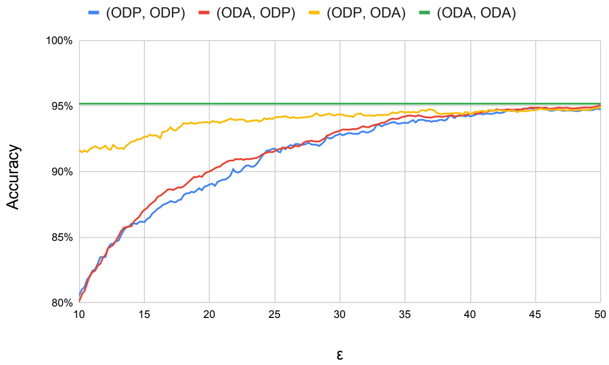


Fig. 3 Experimental comparison of four combinations of (PPTraining, PPTesting) in the WDBC dataset when DR.Rand is used for a dimension reduction. The x-axis is the privacy budget for ODP, and the y-axis is the accuracy.

4.2.4 SUPM.ext data

Figure 3 compares the combination of PPTraining and PPTesting, and the (K, L) values used for each combination are listed below.

- (ODP, ODP) :
  (K, L) ∈ {(5, 2), (7, 4)}
- (ODA, ODP) : (K, L) ∈ {(6, 2), (7, 4)}
- (ODP, ODA) : (K, L) ∈ {(6, 2), (8, 4)}
- (ODA, ODA) : (K, L) ∈ {(7, 2)}

Here, DR.Rand is used to select the attributes. In the training of = ODP, the accuracy is almost the same regardless of the testing data. When comparing (ODA, ODA) and (ODP, ODA), the difference in accuracy was at most 3.57%, showing no significant performance degradation. A similar trend was observed for DR.PW, DR.WA.

4.3 Ionosphere Dataset

The Ionosphere data comprises 351 instances of the radar data collected by the Goose Bay system. This dataset consists 17 pulses with complex values and a label indicating whether the pulses show evidence of some structure in the ionosphere. Because the 17 pulses have complex values and can be split into two real values, this dataset has 34 continuous attributes. However, after we inspected the dataset, we discovered that one of the attributes always has the value 0. An attribute has values of either 0 or 1. Therefore, we consider this dataset to have 32 continuous attributes and one discrete attribute. In the Ionosphere dataset experiments, we used $C = 3.9$ as the parameter for the SVM.

4.3.1 Raw Data

When we directly use this dataset and train an SVM model, we can achieve 95.71% in accuracy. Therefore, the maximum accuracy we can achieve on this dataset is 95.71%.

4.3.2 PW Data

As shown in Figure 5, PW can only achieve 65% accuracy even when using $\epsilon = 50$ as the privacy budget, which is much lower than that acquired using raw data. This result suggests that directly applying PW mechanism to the dataset can significantly degrade the accuracy of the trained models.

4.3.3 SUPM Data

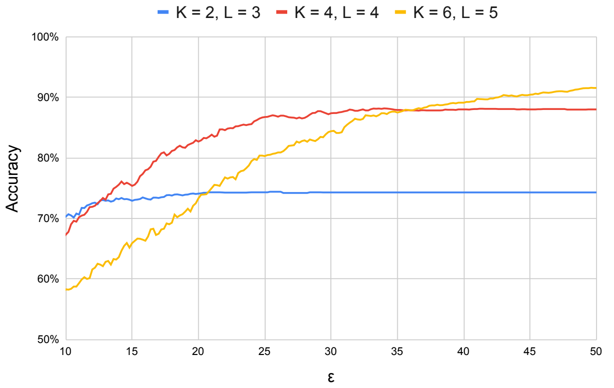


Fig. 4 Accuracy when different values of K and L are configured in the Ionosphere dataset. The x-axis is the privacy budget we applied for training and testing, and the y-axis is the accuracy. ODP data were used for the training and testing phases, and DR.Rand was applied for a dimension reduction.

We experimentally searched for the optimal value of (K, L) and the results indicate that larger K and L values can increase the accuracy when the privacy budget is larger. In contrast, smaller K and L values provide better accuracies when the privacy budget is tighter. Figure 4 illustrates this. Here, we use DR.Rand and train the model using three configurations: $(K, L) = (2, 3)$, $(4, 4)$, and $(6, 5)$. We discovered that when the privacy budget is larger than 36, $(K, L) = (6, 5)$ provides the best result. When ϵ is less than 13, then $(K, L) = (2, 3)$ can maintain the highest accuracy. For the middle range of the ϵ value, we should use $(K, L) = (4, 4)$ instead. Furthermore, we discovered that the same situation appears when we change the dimension-reduction method to DR.PW or DR.WA. To achieve the highest accuracy with our mechanism, we determine the (K, L) values according to the privacy budget, and for the results in Figure 5, we report the highest accuracy when $(K, L) \in \{(2, 2), (3, 2), (4, 2)\}$ is used for DR.PW, $(K, L) \in \{(2, 2), (4, 2)\}$ for DR.WA, and $(K, L) \in \{(2, 3), (4, 4), (6, 5)\}$ for DR.Rand.

When comparing the performances of DR.Rand and DR.WA, DR.WA can provide a considerably higher accuracy when $\epsilon < 25$, but DR.Rand can achieve a slightly higher accuracy when $\epsilon > 47$. DR.WA can provide 85% of accuracy even with a small privacy budget of $\epsilon = 10.2$, while DR.Rand and DR.PW can provide 85% of accuracy when privacy budgets are $\epsilon = 22.8$ and $\epsilon = 41.6$, respectively.

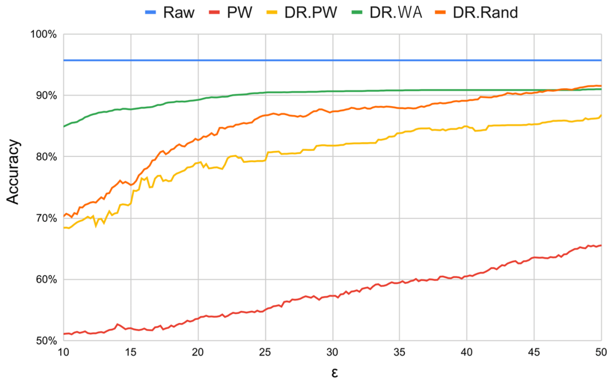


Fig. 5 Experimental comparison of raw, PW, and SUPM data in the Ionosphere dataset. The x-axis is the privacy budget we used during the training and testing phases, and the y-axis shows the accuracy.

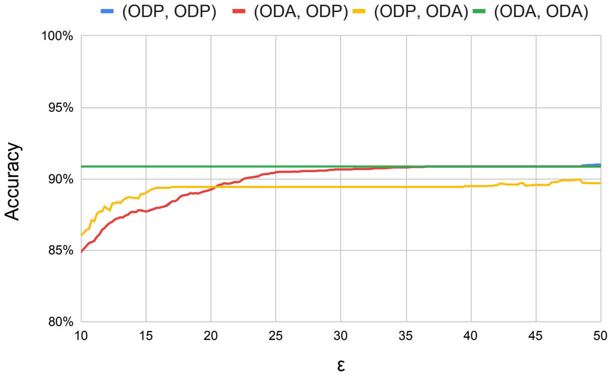


Fig. 6 Experimental comparison of four combinations of (PPTraining, PPTesting) in the Ionosphere dataset when DR.WA was used for a dimension reduction. The x-axis is the privacy budget for ODP, and the y-axis is the accuracy.

4.3.4 SUPM.ext data

Figure 6 compares the four different combinations of (PPTraining, PPTesting), and the (K, L) values used for each combination are listed below.

- (ODP, ODP): $(K, L) \in \{(2, 2), (3, 2), (5, 2)\}$
- (ODA, ODP): $(K, L) \in \{(2, 2), (3, 2)\}$
- (ODP, ODA): $(K, L) \in \{(3, 2), (5, 4), (8, 2)\}$
- (ODA, ODA): $(K, L) \in \{(5, 2)\}$

The combination (ODA, ODA) always provides the best accuracy, because ODA only anonymize the data but not providing differential privacy to them. When $\epsilon < 20$, (ODP, ODA) can provide a better accuracy than (ODP, ODP) and (ODA, ODP). We also discovered that (ODP, ODP) and (ODA, ODP) almost achieves the same accuracy for all ϵ values.

Please note that while we only show the results when we use DR.WA for dimension reduction, we also tested with other dimension reduction methods, and the results were similar.

4.4 Musk Dataset

The Musk dataset consists of 6, 598 instances. Musk dataset is also binary classification between musks or non-musks based on 168 attributes. In the Musk dataset experiments, we used $C = 5.0$ as the SVM parameter.

4.4.1 Raw data

Figure 8 shows that when we directly use the Musk dataset for training and testing, we can achieve an accuracy of 98.98%.

4.4.2 PW data

When the privacy budget ϵ is smaller than 50, the accuracy of the PW data is at most 86.81%, which is lower than SUPM data.

4.4.3 SUPM data

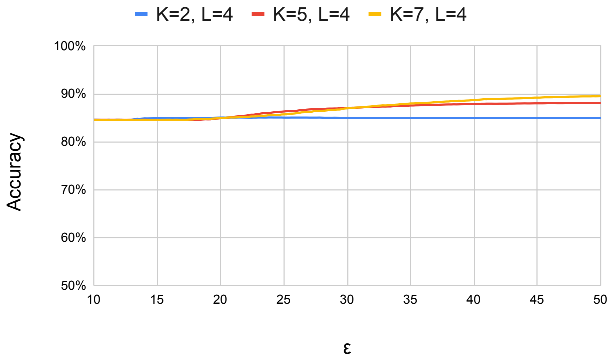


Fig. 7 Accuracy when different values of K and L are configured in the Musk dataset. The x-axis is the privacy budget we used for training and testing, and the y-axis is the accuracy. SUPM peturbed in both the training and testing phases, and DR.Rand was used for a dimension reduction.

Figure 7 shows the results of three different configurations of $(K, L) = \{(2, 4), (5, 4), (7, 4)\}$ with DR.Rand. The greater the number of attributes, the stronger the noise effect is. When the privacy budget is less than 50, the accuracy is reduced when the number of attributes is increased to more than 10. A similar trend was observed for DR.PW, DR.WA.

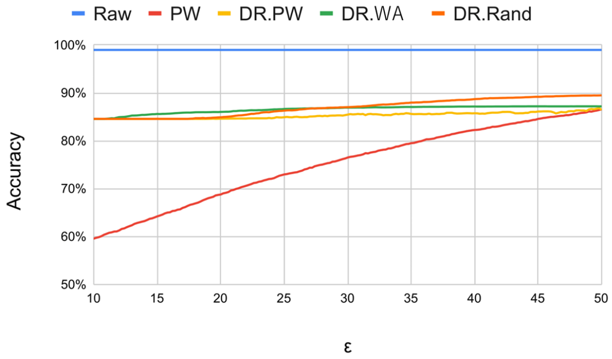


Fig. 8 Experimental comparison of raw, PW, and SUPM-DR.PW, DR.WA and DR.Rand data in the Musk data-set. The x-axis is the privacy budget we applied during the training and testing phases, and the y-axis is the accuracy.

Figure 8 shows the highest accuracy at $(K, L) = \{(3, 4), (5, 4)\}$ for DR.WA, at $(K, L) = \{(2, 4), (5, 4), (7, 4)\}$ for DR.Rand, and at $(K, L) = \{(2, 3), (3, 5), (8, 3)\}$ for DR.PW. It was again shown that by selecting the attributes to which noise is added, it is possible to learn with high accuracy. The Musk dataset shows a smaller noise effect than the other datasets. DR.PW, DR.WA and DR.Rand exceed the accuracy 85% when $\epsilon \geq 27,\ 12.6$, and 20.4, respectively.

4.4.4 SUPM.ext data

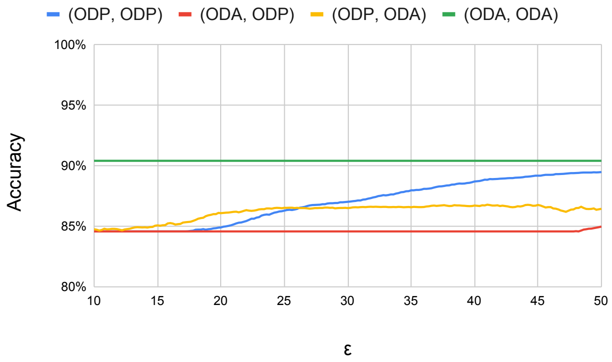


Fig. 9 Experimental comparison of four combinations of (PPTraining, PPTesting) in the Musk dataset when DR.Rand is used for a dimension reduction. The x-axis is the privacy budget for ODP, and the y-axis is the accuracy.

Figure 9 compares the combination of PPTraining and PPTesting, and the (K, L) values used for each combination are listed below.

- (ODP, ODP) : $(K, L) \in \{(2, 4), (5, 4), (7, 4)\}$
- (ODA, ODP) : $(K, L) \in \{(2, 2), (10, 5)\}$
- (ODP, ODA) : $(K, L) \in \{(8, 2), (10, 5)\}$
- (ODA, ODA) : $(K, L) \in \{(10, 2)\}$

Here, DR.Rand is used to select the attributes. Accuracy of (ODP, ODP), (ODA, ODP), and (ODP, ODA) are almost the same if $\epsilon < 15$ although that of (ODP, ODA) is slightly higher than the other two. These experimental results indicate that, as with the WDBC and Ionosphere datasets, adding noise to the training data does not result in a significant degradation in the performance.

5. Analysis

There are two effective methods when building a privacy-preserving machine learning model, i.e., applying a privacy-preserving technique such as an LDP mechanism to the data and incorporating a privacy-preserving mechanism when building a learning model. For the former, a mechanism applied directly to the data was found to be experimentally inefficient. The problem is that LDP mechanisms, such as PW, are primarily intended for statistical analysis, and the data-specific characteristics of the data are largely destroyed. Although many studies have reported on the latter, they are specialized to a particular use case and are therefore not general-purpose. Herein, we proposed a framework for constructing highly accurate machine learning models by applying weak data anonymization to reduce excessive information. We achieved several different findings throughout the experiments. The confirmed trends are similar to the results of experiments conducted with simple datasets in our previous study [12].

5.1 Choosing K and L

When determining the values K and L during a dimensionality reduction for SUPM data, we discovered that when we have a higher privacy budget, we should spend it on more attributes rather than reducing the noise applied to the data. This is because when the level of noise is lower than a certain magnitude, reducing it further does not enhance the accuracy of our models, whereas adding more attributes can help the model achieve a higher accuracy. Meanwhile, when we want to tighten the privacy budget, using too many attributes forces us to impose significant noise on the data, causing the accuracy to decrease. Therefore, reducing the number of attributes allows us to apply lower-scale noise to the data, and a higher accuracy can be achieved. This also occurs with the number of classes because they can provide a higher resolution when there are more classes. However, the scale of the noise is also increased. In the future, establishing an optimal K and L discovery method will enable us to construct an optimal model according to the required privacy level.

5.2 Dimension Reduction Comparison

By comparing our dimension reduction methods, we determined that while providing the same level of privacy, DR.Rand also offers a higher accuracy than DR.PW in most cases. This is because DR.PW must occupy a certain privacy budget when calculating the correlation coefficients, and the increased noise degrades model performance. This feature can be reversed depending on the number of attributes and data applied in the dataset. When comparing DR.WA and DR.Rand, DR.WA was found to consistently perform better than DR.Rand. However, because DR.WA only provides weak privacy protection for sensitive data, using DR.Rand can provide differential privacy protection to the dataset and thus, from a privacy perspective, is a better method.

6. Conclusion

Because a record can consist of both continuous and discrete data, this study proposed a privacy mechanism ODP to handle data uniformly regardless of the data type. ODP controls two axes of privacy and accuracy using the number of classes of one data as well as the privacy budget. We also proposed a privacy-preserving machine learning framework, SUPM, to control the entire set of data used in all phases of dimension-reduction, training, and testing. To the best of our knowledge, this is the first privacy-preserving machine learning framework that focuses on both data privacy and model privacy and enables all phases of dimension-reduction, training, and testing without assuming TTPs or trusted clients.

References

[1] C. Gentry, “Fully homomorphic encryption using ideal lattices,” in *Proceedings of the forty-first annual ACM symposium on Theory of computing*, 2009, pp. 169–178.

[2] D. Demmler, T. Schneider, and M. Zohner, “Aby-a framework for efficient mixed-protocol secure two-party computation.” in *NDSS*, 2015.

[3] M. Bellare, V. T. Hoang, and P. Rogaway, “Foundations of garbled circuits,” in *Proceedings of the 2012 ACM conference on Computer and communications security*, 2012, pp. 784–796.

[4] P. Xie, M. Bilenko, T. Finley, R. Gilad-Bachrach, K. Lauter, and M. Naehrig, “Crypto-nets: Neural networks over encrypted data,” *arXiv preprint arXiv:1412.6181*, 2014.

[5] C. Dwork, “Differential privacy,” in *Proc. of ICALP 2006, LNCS*, vol. 4052, 2006, pp. 1–12.

[6] P. Kairouz, K. Bonawitz, and D. Ramage, “Discrete distribution estimation under local privacy,” in *International Conference on Machine Learning*. PMLR, 2016, pp. 2436–2444.

[7] T. Wang, J. Blocki, N. Li, and S. Jha, “Locally differentially private protocols for frequency estimation,” in *26th USENIX Security Symposium (USENIX Security 17)*, 2017, pp. 729–745.

[8] M. Gaboardi and R. Rogers, “Local private hypothesis testing: Chi-square tests,” in *International Conference on Machine Learning*, 2018, pp. 1626–1635.

[9] B. Ding, H. Nori, and e. Li, “Comparing population means under local differential privacy: with significance and power,” in *Proceedings of the AAAI*, vol. 32, no. 1, 2018.

[10] K. Wei, J. Li, M. Ding, C. Ma, H. H. Yang, F. Farokhi, S. Jin, T. Q. Quek, and H. V. Poor, “Federated learning with differential privacy: Algorithms and performance analysis,” *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3454–3469, 2020.

[11] N. Wang, X. Xiao, Y. Yang, J. Zhao, S. C. Hui, H. Shin, J. Shin, and G. Yu, “Collecting and analyzing multidimensional data with local differential privacy,” in *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE, 2019, pp. 638–649.

[12] A. Miyaji, T. Takahashi, P.-L. Wang, T. Yamatsuki, and T. Mimoto, “Privacy-preserving data analysis without trusted third party,” in *2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. IEEE, 2022, pp. 710–717.

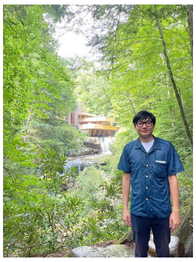
[13] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, “Scikit-learn: Machine learning in python,” *the Journal of machine Learning research*, vol. 12, pp. 2825–2830, 2011.

[14] J. C. Duchi, M. I. Jordan, and M. J. Wainwright, “Local privacy and statistical minimax rates,” in *2013 IEEE 54th annual symposium on foundations of computer science*. IEEE, 2013, pp. 429–438.
[15] P. Kairouz, S. Oh, and P. Viswanath, “Extremal mechanisms for local differential privacy,” *Advances in neural information processing systems*, vol. 27, 2014.
[16] B. I. Rubinstein, P. L. Bartlett, L. Huang, and N. Taft, “Learning in a large function space: Privacy-preserving mechanisms for svm learning,” *arXiv preprint arXiv:0911.5708*, 2009.
[17] “Breast cancer wisconsin (diagnostic) data set,” UCI Machine Learning Repository https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(diagnostic).
[18] “Ionosphere data set,” UCI Machine Learning Repository https://archive.ics.uci.edu/ml/datasets/ionosphere.
[19] N. Holohan, D. J. Leith, and O. Mason, “Optimal differentially private mechanisms for randomised response,” *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 11, pp. 2726–2735, 2017.
[20] “Musk(version2) data set,” UCI Machine Learning Repository https://archive.ics.uci.edu/ml/datasets/Musk+(Version+2).

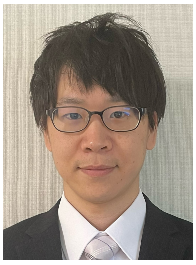
Atsuko Miyaji received the B. Sc., the M. Sc., and the Dr. Sci. degrees in mathematics from Osaka University,in 1988, 1990, and 1997 respectively. She is an IPSJ fellow. She was an associate professor at the Japan Advanced Institute of Science and Technology (JAIST) in 1998. She was a professor at Japan Advanced Institute of Science and Technology (JAIST) from 2007 to 2023. She has been a professor at Graduate School of Engineering, Osaka University since 2015. She received Young Paper Award of SCIS’93 in 1993, Notable Invention Award of the Science and Technology Agency in 1997, the IPSJ Sakai Special Researcher Award in 2002, the Standardization Contribution Award in 2003, the AWARD for the contribution to CULTURE of SECURITY in 2007, the Director-General of Industrial Science and Technology Policy and EnvironmentBureau Award in 2007, DoCoMo Mobile Science Awards in 2008, Advanced Data Mining and Applications (ADMA 2010) Best Paper Award, Prizes for Science and Technology, the Commendation for Science and Technology by the Minister of Education, Culture, Sports, Science and Technology, International Conference on Applications and Technologies in Information Security (ATIS 2016) Best Paper Award, the 16th IEEE Trustocm 2017 Best Paper Award, IEICE milestone certification in 2017, the 14th Asia Joint Conference on Information Security (AsiaJCIS 2019) Best Paper Award, Information Security Applications - 20th International Conference (WISA 2020) Best Paper Gold Award, IEICE Distinguished Educational Practitioners Award in 2020, and IEICE Achievement Award in 2023. She is a member of the International Association for Cryptologic Research, the Institute of Electrical and Electronics Engineers, the Institute of Electronics, Information and Communication Engineers, the Information Processing Society of Japan, the Mathematical Society of Japan, the Japanese Society for Artificial Intelligence, and Japan Association for Medical Informatics.

Tatsuhiro Yamatsuki received the B. Sc., in engineering from Nara Osaka University in 2022, and received the M. Sc., degree in engineering from Osaka University in 2024.

Tomoka Takahashi received the B. Sc., in mathematics from Nara Women’s University in 2021, and received the M. Sc., degree degree in engineering from Osaka University in 2023. She has been with MegaChips Corporation since April 2023.



Ping-Lun Wang is a Ph.D. student at Carnegie Mellon University. His current research is about hardware security and side channels inside a processor. Before starting his Ph.D., he was an exchange student at Osaka University, and he received his M.S. and B.S. at National Taiwan University.



Tomoaki Mimoto received B.E. and Ph.D degrees from Osaka university, Japan, in 2012 and 2022, and received M.E. (Outstanding Performance Award) in information science from Japan Advanced Institute of Science and Technology in 2014. He joined KDDI in 2014, and was with KDDI research, Inc. from 2015 to 2020. In 2020, he moved to Advanced Telecommunications Research Institute International (ATR). Since 2023, he is a researcher in KDDI Research, Inc. again.