Journal of Database Management
Volume 33 • Issue 2

Random Forest Algorithm Based on Linear Privacy Budget Allocation

Yanling Dong, North China University of Science and Technology, China
Shufen Zhang, North China University of Science and Technology, China*
Jingcheng Xu, North China University of Science and Technology, China
Haoshi Wang, North China University of Science and Technology, China
Jiqiang Liu, Beijing Jiaotong University, China

ABSTRACT

In the era of big data with exponential growth in data volume, how to reduce data security issues such as data leakage caused by machine learning is a hot area of recent research. The existing privacy budget allocation strategies are usually only suitable for data applications in specific spaces and cannot meet users' personalized needs for privacy budget allocation. Therefore, a linear privacy budget allocation strategy is proposed. The strategy assigns each layer a linearly increasing privacy budget from the root of the decision tree to the bottom by adjusting the coefficient or constant term. Combining this strategy with the random forest algorithm, a random forest algorithm based on linear privacy budget allocation (DiffPRF_linear) is formed. Experimental results show that the proposed algorithm can realize uniform, arithmetic, and geometric privacy budget allocation policy effects and can also achieve better classification effects than the former, which not only meets the needs of users to protect private data personalized but also maintains high classification accuracy.

KEYWORDS

Data Security, Differential Privacy, Machine Learning, Privacy Budget Allocation, Privacy Protection, Random Forest

INTRODUCTION

Currently, big data technology is more deeply and widely used (Wang et al., 2013; Tao et al., 2013), and artificial intelligence technologies such as machine learning and deep learning are also accelerating development (Wang et al., 2019). To obtain more convenient digital services, a "portrait" based on the personal data of group users is inevitable. However, the "portrait" in different fields has different requirements for data collection, combined with the different security levels of data storage by data collectors and the strong background knowledge of attackers, which makes data leakage, data selling

DOI: 10.4018/JDM.309413 *Corresponding Author

and other data security problems (Wang et al., 2019) pose significant risks to users, developers and society (Siau et al., 2020). Aiming at reducing the risk of privacy disclosure (Dumbill et al., 2013; Meng et al., 2013), researchers have proposed privacy protection technologies such as anonymization technology (Sweeney, 2002; Machanavajjhala et al., 2007; Li et al., 2007; Xiao et al., 2007), cryptography technology (Clifton et al., 2002; Rothe, 2002; Jiang et al., 2006; Ishai et al., 2006), differential privacy technology (Dwork, 2006; Dwork, 2008; Dwork et al., 2009) and blockchain technology (Turesson et al., 2021). Among them, differential privacy technology is currently the mainstream privacy protection technology. It is difficult for attackers to use background knowledge to predict the sensitive attributes of individuals who add noise compared to the primary data, so as to control the disclosure of personal privacy in a small range (Li et al., 2012; Xiong et al., 2104).

Adjusting the magnitude of the privacy budget can offer different degrees of protection for data and affect data availability (Mcsherry et al., 2007). Dwork et al. (2012) first proposed a uniform allocation strategy to distribute the privacy budget, i.e., the privacy protection budget is evenly distributed to each layer of the tree structure. However, such an allocation scheme will lead to more waste of the privacy budget, low use efficiency of the privacy budget, and the allocation of the privacy budget is relatively fixed and unadjustable. Cormode et al. (2012) proposed a geometric budget allocation strategy, namely increasing the privacy budget geometrically from the root node, but the query error may be proportional to the number of leaves in the query. Wang et al. (2016) put forward an adaptive privacy budget allocation strategy to provide privacy protection statistics which is published on unlimited timestamps, but this strategy can only be applied to specific data. Wang (2019) proposed a p-series privacy budget allocation method for infinite attacks that attackers may launch. However, when there are too many iterations, the privacy budget tends to be zero, resulting in poor data availability. In the real-time location protection of users, Li et al. (2017) proposed a privacy budget allocation strategy with adaptive adjustment according to the distribution of underlying data. However, when the total number of users is too large, the privacy budget allocated on each timestamp is too small, resulting in the addition of too much noise. Wang et al. (2018) proposed the privacy budget allocation method of arithmetic sequences and geometric sequences, but it is vulnerable to the influence of parameters and the privacy budget allocation method is not flexible enough. Ke et al. (2021) proposed a dynamic privacy allocation mechanism on the basis of the different output contributions of different input features to the model, while the accuracy of the model is low and the balance between model utility and privacy protection is not reached.

Blum et al. (2005) first combined differential privacy technology with a single learner, proposed the SuLQ-based ID3 algorithm, and applied Laplace to add noise. However, due to the excessive noise added, the accuracy of the classification results decreased significantly compared with that without noise. Since a single individual learner can no longer satisfy people's desire to have a stable model with good performance in all aspects, Breiman (2001) proposed the random forest algorithm. Patil et al. (2014) first combined the random forest algorithm with differential privacy and used the ID3 algorithm to construct the subtree of the random forest. However, this method can only handle discrete attributes, and can only preprocess continuous features when facing continuous features. The DiffPRFs algorithm proposed by Mu et al. (2016) eliminates the dependence on discrete datasets and selects split points and split attributes through the exponential mechanism. However, each iteration requires two calls to the exponential mechanism, which consumes too much privacy budget. Li et al. (2020) further proposed the RFDPP-Gini algorithm, using the CART classification tree as a subtree and invoking the exponential mechanism only once when processing continuous features to improve the use efficiency of the privacy protection budget. However, since the privacy budget allocation scheme selected is uniform allocation, the utilization rate of the privacy budget is low.

The authors present DiffPRF_linear, a stochastic forest algorithm based on linear privacy budget allocation. Through a linear allocation strategy, users can more flexibly adjust the coefficient or constant term to allocate the privacy budget of each layer, and can also realize the effect of uniform,

arithmetic and geometric privacy budget allocation strategy. In addition, the random forest algorithm retains high classification accuracy while protecting personal privacy.

Differential privacy protection technology

This technology was first applied in the field of database security. Attackers cannot carry out differential attacks when querying two datasets with only one record difference and the probability of obtaining the same result is very similar.

Definition and Related Concepts of Differential Privacy

Let dataset D and dataset D' have the same attribute structure, $D\Delta D'$ is called the difference of symmetry between the two, and $\left|D\Delta D'\right|$ represents the number of records in $D\Delta D'$. If $\left|D\Delta D'\right|=1$, D and D' are called adjacent datasets.

Definition 1. Differential Privacy (Dwork, 2006). Given a random algorithm M, where PM is the set of all possible outputs of M. For any two adjacent datasets D and D' and any subset SM of PM, if algorithm M satisfies:

$$\Pr[M\left(D\right)\in SM\left]\leq e^{\varepsilon}\Pr[M\left(D'\right)\in SM\right] \tag{1}$$

i.e, algorithm M provides differential privacy protection, where parameter ε is called privacy protection budget, which is used to ensure the probability of consistent output results of algorithm M by adding or reducing one record in the dataset, and $Pr[]$ represents the probability of occurrence of an event.

Definition 2. Local sensitivity (Yang et al., 2019). Given any query function $f:D\rightarrow R^{d}$, for the adjacent datasets D' of given datasets D and D, the local sensitivity of f on D is $LS_{f}=max_{D'}\left\|f\left(D\right)-f\left(D'\right)\right\|_{1}$, where $\left\|\cdot\right\|_{1}$ stands for Manhattan distance.

Differential privacy protection technology has flexible combination characteristics, which are serial combination and parallel combination.

Definition 3. Serial Combination (Dwork et al., 2006). Given that there are n random algorithms K, where K_{i} satisfies ε_{i}-differential privacy, then for the same dataset, the algorithm combined by $\left\{K_{i}\right\}\left(1¡¦i¡¦n\right)$ satisfies $sum\left(\varepsilon_{i}\right)$-differential privacy.

This characteristic shows that in the same dataset, if each algorithm satisfies ε_{i}-differential privacy, then the combined algorithm composed of these algorithm sequences will provide the privacy protection ability of the sum of the privacy budgets of all algorithm sequences.

Definition 4. parallel Combination (Dwork et al., 2006). Given that there are n random algorithms K, where K_{i} satisfies ε_{i}-differential privacy, and any two datasets do not intersect, then the algorithm combined by $\left\{K_{i}\right\}\left(1¡¦i¡¦n\right)$ satisfies $max\left(\varepsilon_{i}\right)$-difference privacy.

This characteristic shows that in a dataset providing differential privacy protection algorithms, if the subdata sets processed by each algorithm do not intersect each other, then the privacy protection level provided by the combined algorithm composed of these algorithm sequences depends on the algorithm sequence with the worst protection level, namely the largest budget.

Mechanism

Differential privacy technology usually realizes data privacy protection through the Laplace mechanism and exponential mechanism. For numerical data, protection is usually applied by the Laplacian mechanism, and for nonnumerical data, protection is usually applied by Exponential mechanism.

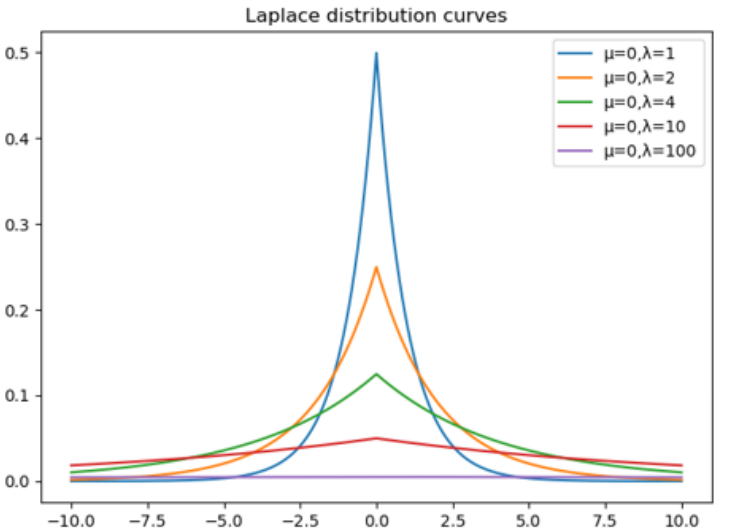
The Laplace mechanism realizes differential privacy protection by adding random variable noise to query the results, and the added random variable noise obeys the Laplace distribution. Given that the scale parameter is λ and the position parameter is 0, the Laplacian distribution is $Lap\left(\lambda\right)$ and its probability density function is:

$$P\left(x\right)=\frac{1}{2\lambda}\cdot e^{-\frac{|x|}{\lambda}} \tag{2}$$

The Laplacian distribution of parameter λ at different scales is shown in Figure 1

Definition 5. Laplace Mechanism (Dwork et al., 2006). Given dataset D and privacy budget ε, with query function $f\left(D\right)$, whose sensitivity is Δf, then the random algorithm $M\left(D\right)=f\left(D\right)+Y$ offers $\varepsilon-$ differential privacy, where $Y\sim Lap\left(\Delta f/\varepsilon\right)$ is random noise. It obeys the Laplacian distribution with scale parameter $\Delta f/\varepsilon$. The overall idea of the Laplace mechanism is to output outliers with a certain probability. To make the noisy data deviate from the real value, it is necessary to improve the probability of outlier output, i.e., to stably obtain an outlier. Therefore, when the Laplace function curve is flatter, the probability of outlier output is higher.

Figure 1. Laplacian distributions of parameter λ at different scales



Different from the Laplace mechanism, the Exponential mechanism achieves differential privacy by not simply adding noise to the query results but introducing a scoring function to obtain a score for each possible output, which is normalized as the probability value returned by the query, i.e., the element in the discrete data $\{R_1, R_2, \ldots, R_N\}$ is output probabilistically.

Definition 6. Exponential Mechanism (Mcsherry, 2007). Given the random algorithm M, the input dataset is D, the output is an entity object $R_i \in \{R_1, R_2, \ldots, R_N\}$, and $\mathrm{q}\left(\mathrm{D}, \mathrm{R}_\mathrm{i}\right)$ is the availability function to evaluate the quality of the output value R_i, where Δq is the sensitivity of function $\mathrm{q}\left(\mathrm{D}, \mathrm{R}_\mathrm{i}\right)$. If algorithm M selects and outputs R_i from R_i ¡¡§°{R_1, R_2,¡-, R_N} with probability proportional to $e^{\frac{\varepsilon q(D,r)}{2\Delta q}}$, then algorithm M provides ε -differential privacy protection. The overall idea of the exponential mechanism is that, after receiving a query, it does not output a result deterministically, but uses a scoring function to measure the probability that the output value R_i is selected, to achieve differential privacy. Among them, the higher the scoring function value is, the higher the probability of being selected for output; the lower the value is, the lower the probability of being selected for output.

RANDOM FOREST

Random forest, proposed by Breiman (1999), is an ensemble learning algorithm that combines several weak learners into strong ones, and its base learner is a decision tree. In this paper, the CART classification tree is used as the base decision tree in random forest.

CART (Classification and Regression Tree) Decision Tree

The CART tree is a binary tree that handles both classification and regression. Fearn et al. (2006) used square error minimization criterion to generate a regression tree for continuous data. For discrete data, the Gini coefficient minimization criterion is used to generate a classification tree. The smaller the Gini coefficient is, the more significant the feature is.

Definition 7. Gini coefficient (Breiman, 1984). Assuming that there are n categories and the probability of the i-th category is p_i, the Gini coefficient of the probability distribution can be expressed as:

$$Gini\left(p\right) = \sum_{i=1}^{n} p_i \left(1 - p_i\right) = 1 - \sum_{i=1}^{n} p_i^2 \tag{3}$$

For the dichotomy problem, if p is the probability of the sample belonging to the first category, the Gini coefficient of the probability distribution can be expressed as:

$$Gini\left(p\right) = 2p\left(1 - p\right) \tag{4}$$

For sample set D, which counts $\left|D\right|$, and the i-th category counts $\left|C_i\right|$, the Gini coefficient expressions for sample set D can be expressed as:

$$Gini\left(D\right)=1-\sum_{i=1}^{n}\left(\frac{\left|C_i\right|}{\left|D\right|}\right)^2 \tag{5}$$

In light of a certain value a of feature A, D is divided into D_1 and D_2. Then under the condition of feature A, the Gini coefficient of sample D can be expressed as:

$$Gini\left(D,A\right)=\frac{\left|D_1\right|}{\left|D\right|}Gini\left(D_1\right)+\frac{\left|D_2\right|}{\left|D\right|}Gini\left(D_2\right) \tag{6}$$

CART Classification Tree Algorithm

In practice, each dataset sample usually includes continuous attributes and discrete attributes. The CART classification tree has different processing methods for different types of attributes.

Continuous Attribute Processing by the CART Classification Tree Algorithm

When facing continuous attributes, continuous attributes need to be discretized. Assuming that there are m samples in the dataset, the continuous feature A has m values, which are sorted from small to large, and the average value of two adjacent samples is taken as the dividing point. There are $m-1$ dividing points in total, and the i-th dividing point T_i is represented as $T_i=\left(a_i+a_{i+1}\right)/2$. The Gini coefficients of the $m-1$ points were calculated respectively, and the point with the lowest Gini coefficient was selected as the binary discrete classification point with the continuous feature. Then, the dataset of this node is divided into two parts on the basis of this point.

Discrete Attribute Processing by the CART Classification Tree Algorithm

When facing discrete attributes, the Gini coefficient of each value of the existing discrete attributes of the current node is first calculated, and the feature with the smallest Gini coefficient and its corresponding value are selected as the optimal feature and optimal split point. Then, the dataset of this node is divided into two parts on the basis of this point.

Algorithm 1. CART classification tree generation algorithm
Input: Training dataset D, Gini coefficient threshold Thr_gini, sample number threshold Thr_sample, feature set A.
Output: CART classification tree.
1: Initial Thr_gini, Thr_sample.
2: Starting from the root node, recursively perform the following for each node, until the number of node samples is less than Thr_sample, or the Gini coefficient of the sample set is less than Thr_gini.
3: Let the training dataset of the node be D_n.
4: **for** each feature in A
5: For each possible value a, calculate the Gini coefficient when $A_i=a$.
6: Take the minimum Gini coefficient as the optimal splitting point of A_i.
7: **end for**

8: Take the feature of the smallest Gini index and its corresponding eigenvalue, and the training dataset is divided into two subnodes according to the feature and eigenvalue.

Random Forest Algorithm

The core of random forest is sample randomness and attribute randomness. During classification, each subtree outputs the classification with the most votes through "voting" as the final result.

Algorithm 2. Random forest generation algorithm based on the CART classification tree

Input: Training dataset D, Number of CART classification trees T, Depth of CART classification tree h, number of features selected randomly during splitting m, minimum threshold of splitting species Thr, and feature set F.

Output: Random Forest composed of T CART classification trees.

1: Initial T, h, m, Thr.
2: **for** $i=1$ **to** T
3: Randomly draw N training samples with replacement from the training dataset D to obtain the training set D_t.
4: **if** (Reach the minimum threshold of split species Thr or reach the maximum depth of decision tree h)
5: Generate CART classification tree T_i.
6: **else**
7: Randomly select m features in feature set F, and select the optimal split feature as the node.
8: **end if**
9: **end for**

DESIGN AND DESCRIPTION OF THE RANDOM FOREST ALGORITHM BASED ON LINEAR PRIVACY BUDGET ALLOCATION

Linear Privacy Budget Allocation Strategy

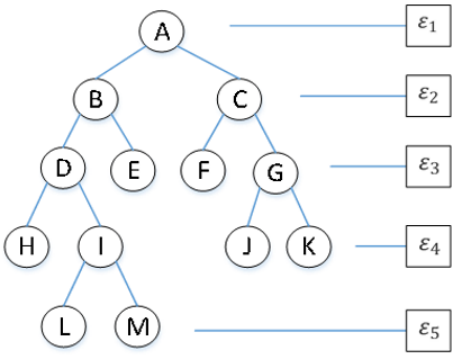
For the sake of meeting the needs of users to be able to individually adjust the privacy budget and reasonably protect personal privacy, this paper puts forward a linear allocation of the privacy budget strategy. This article takes a full binary tree (Cormen et al., 2001) as an example, assuming that the depth of the tree is h, where the root node resides is the first layer, and the leaf node is the i-th layer, as shown in Figure 2.

From the level where the root node is located, the privacy budget of $\varepsilon_1, \varepsilon_2, \varepsilon_3, \cdots, \varepsilon_h$ is allocated to each layer of the decision tree with depth h. From the second layer, the privacy budget allocated to each layer is:

$$\varepsilon_i = \varepsilon_1 q^{i-1} + (i-1)d; \ \ (i>1) \tag{7}$$

Where q^{i-1} is the coefficient $(q \geq 1)$ and $(i-1)d$ is the constant term $(d \geq 0)$, which can be known from the serial combination of property (1) in the nature of differential privacy combination:

Figure 2. Full binary tree



$$\varepsilon = \sum_{i=1}^{h} \varepsilon_i = \frac{2\varepsilon_1\left(1-q^h\right)+\left(h-1\right)\left(1-q\right)hd}{2\left(1-q\right)};\left(q>1\right) \tag{8}$$

$$\varepsilon = \sum_{i=1}^{h} \varepsilon_i = h\varepsilon_1 + \frac{d\left(h-1\right)h}{2};\left(q=1\right) \tag{9}$$

It can be obtained from equation (8) and equation (9):

$$\varepsilon_1 = \frac{2\left(1-q\right)\varepsilon-\left(h-1\right)\left(1-q\right)hd}{2\left(1-q^h\right)};\left(q>1\right) \tag{10}$$

$$\varepsilon_1 = \frac{2\varepsilon-hd\left(h-1\right)}{2h};\left(q=1\right) \tag{11}$$

Because $\varepsilon_1 > 0$, then $0 \le d < \frac{2\varepsilon}{h\left(h-1\right)}$.

Substituting Formula (7) into Formula (10) and Formula (11) respectively, the privacy protection budget allocated to each node in the i-th layer $(i>0)$ of the decision tree is:

$$\varepsilon_i = \begin{cases} \left[\frac{2\left(1-q\right)\varepsilon-\left(h-1\right)\left(1-q\right)hd}{2\left(1-q^h\right)}\right]q^{i-1}+\left(i-1\right)d\ ; & \left(q>1, 0\le d<\frac{2\varepsilon}{h\left(h-1\right)}\right) \\ \frac{2\varepsilon-hd\left(h+1-2i\right)}{2h}\ ; & \left(q=1, 0\le d<\frac{2\varepsilon}{h\left(h-1\right)}\right) \end{cases} \tag{12}$$

DiffPRF_linear Algorithm Description

The linear privacy budget allocation strategy has excellent combinability. Multiple privacy budget allocation schemes can be obtained by combining coefficient q and constant d. Therefore, the DiffPRF_linear algorithm is proposed based on the combination of a linear privacy budget allocation strategy and a random forest algorithm to realize users' personalized demand for privacy protection.

Algorithm 3. DiffPRF_linear Generation Algorithm

Input: Training dataset D, decision trees count T, depth of decision tree h, features selected randomly during splitting count m, total privacy protection budget B, parameter q, parameter d, feature set F.

Output: Random Forest satisfying ε-differential privacy.

1: Initial T, h, B, m, d, q.

2: Calculate the privacy budget $\varepsilon_T = B/T$ of each tree.

3: Calculate the privacy budget ε_1 of each node in the first layer of the decision tree.

4: Calculate the privacy budget ε_i of each node in the second to h layers of the decision tree, and divide it into two equal parts $\varepsilon_N = \varepsilon_i / 2$.

5: **for** $i = 1$ **to** T

6: Randomly draw n training samples with replacement from D to obtain the training set D_n.

7: Recursively execute the following steps to build a decision tree T_i.

8: **if** (The classification of all samples in the node is consistent, or the samples are smaller than the minimum number of samples required for splitting, or B is exhausted, or reach h)

9: Select the category with the largest value in the sample as the label of the leaf node.

10: **else**

11: Set the current node as a decision node.

12: Calculate the number of samples for each category of the decision node, and add Laplace noise.

13: **if** (D_n has a continuous feature)

14: Randomly select several features from F.

15: **if** (There are m continuous features)

16: Allocate the privacy budget equally to each continuous feature and one copy is reserved for discrete features $\varepsilon' = \varepsilon_N / (m+1)$.

17: Use the following formula to calculate the value of the scoring function, and take the highest value as the best continuous feature and split point. //$1 - \dfrac{\exp\left(\dfrac{\varepsilon'}{2\Delta q} q(D_c, F)\right)}{\sum_i \exp\left(\dfrac{\varepsilon'}{2\Delta q} q(D_c, F)\right)}$; q is the Gini coefficient, the smaller the better, Δq is the sensitivity of the Gini coefficient, and D_c is the training set

```
on the current node.
18:          Calculate the Gini coefficient of remaining discrete
feature and compare it with the best feature in the continuous
feature. Take the smallest Gini coefficient as the optimal split
feature and optimal split point, and split this node.
19:      else
20:           Calculate the Gini coefficient of discrete
features, take the smallest Gini coefficient as the optimal split
feature and optimal split point, and split the node according to
this feature and eigenvalue.
21:      end if
22:    else
23:       Calculate the Gini coefficient of discrete features,
take the smallest Gini coefficient as the optimal split feature
and optimal split point, and split the node according to this
feature and eigenvalue.
24:    end if
25:   end if
26: end for
```

After establishing a random forest satisfying ε -differential privacy through Algorithm 3, the description of the process of classifying the test set is shown in Algorithm 4.

Algorithm 4. Algorithm for classifying test set samples with random forest satisfying ε -differential privacy

```
Input: Test set S, random forest satisfying ε-differential
privacy.
Output: Test set classification results.
1: for each sample in S.
2:   for i=1 to T.
3:     Starting from the root node of the i-th tree, on the basis
of the classification result of the current node, determine which
branch node the sample enters until it reaches a certain leaf
node.
4:     Obtain the classification result of the i-th tree.
5:   end for
6:   Take the mode of the classification results of all trees as
the classification results of the test set.
7: end for
```

Algorithm Analysis

All the privacy budgets consumed in the DiffPRF_linear algorithm conform to ε -differential privacy. The total privacy budget is evenly distributed to each decision tree, $\varepsilon_T = B / T$, and a linearly increasing privacy budget is distributed to each layer from the top to the bottom according to the linear distribution method, i.e., $\varepsilon_i = \varepsilon_1 q^{i-1} + (i-1)d \quad \left\{0 \le d \left\langle \frac{2\varepsilon_T}{h(h-1)}, i \right\rangle 1\right\}$. Users only need to adjust the value of d or q to change the size of each layer's privacy budget, and can achieve the needs of uniform, arithmetic, and geometric privacy budget allocation. When $d = 0$ and $q = 1$, it is equivalent to the uniform distribution method, i.e., the privacy budget allocated at each layer of

the tree is equal. When $q=1$, $\varepsilon_i=\varepsilon_1+(i-1)d;\left\{0\le d\left\langle\frac{2\varepsilon}{h(h-1)},i\right\rangle 1\right\}$ is equivalent to the arithmetic distribution method, that is, the privacy budget allocated at each layer from root to leaf nodes increases with constant d. When $d=0$, $\varepsilon_i=\varepsilon_1 q^{i-1}$;$(q>1)$, which is equivalent to the geometric allocation method, i.e., the privacy budget allocated at each layer from root to leaf nodes increases by a constant d times. Therefore, the analysis shows that the linear privacy budget allocation strategy is more flexible and has more portfolio diversity.

EXPERIMENT AND ANALYSIS

The Experimental Data

The programming language used in this experiment is Python 3.8. The selected datasets are derived from the Adult and Bank Marketing datasets from the publicly available UCI machine learning database. There are 32561 data samples in the Adult dataset, and each sample contains 15 attributes, including 14 classification features and 1 classification result. Through the analysis of the attribute set, it was found that there were 5057 samples of non-American nationality, accounting for only 16% of the total sample number. Therefore, only the sample records of American nationality were selected for training and testing in this experiment, which included 6 continuous attributes and 7 discrete attributes. The classification attribute 'income level' can be divided into "$\le 50k$" and "$>50k$". There are 41188 data samples in the Bank Marketing dataset, and each sample has 21 attributes, including 20 classification features and 1 classification result. Through the analysis of the attribute set, it was found that the sample attribute contained 10 continuous attributes and 10 discrete attributes. The classification attribute 'has the client subscribed a term deposit' was divided into two categories: "Yes" and "no". In the following experiment, 70% of the Adul and Bank Marketing datasets were used as training sets and 30% were used as test sets.

Experimental Results and Analysis

When ε_T = 0.2, 0.4, 0.8, 1, 2, 4, 8, T = 25, h=9, the classification accuracy of the DiffPRF_linear algorithm was calculated under different combinations of parameter q and parameter d .Each group of experiments with the same parameter was run repeatedly 15 times and the average value was obtained, as shown in Figure 3 and Figure 4. $d_1=0.25\frac{2\varepsilon_T}{h(h-1)}$, $d_2=0.5\frac{2\varepsilon_T}{h(h-1)}$, $d_3=0.75\frac{2\varepsilon_T}{h(h-1)}$, $q_1=1.1$, $q_2=1.2$, $q_3=1.3$ and $q_4=2$.

It can be seen from Figures 3 and 4, when ε_T increases, the classification accuracy of the DiffPRF_linear algorithm decreases to varying degrees, but the overall trend is rising. From Figs. 3(a) ~ 3(c) and Figs. 4(a) ~ 4(c), it can be seen that when $\varepsilon_T\le 0.8$, the classification accuracy of the DiffPRF_linear algorithm shows a positive increase, and the growth rate is faster. When $\varepsilon_T>0.8$, the classification accuracy of the DiffPRF_linear algorithm occasionally increases negatively, but tends to gently increase. This is because when ε_T is too small, the privacy budget allocated to the node is extremely small. At this time, $\Delta f/\varepsilon$ is extremely large, i.e., adding excessive noise. On the basis of the Laplace function curve in Figure 1, when $\lambda=\Delta f/\varepsilon$ is too large, the curve tends to a straight line, and the probability of outlier output is extremely high, resulting in extremely low classification accuracy of the algorithm. When ε_T gradually increases, $\Delta f/\varepsilon$ gradually decreases,

i.e., the added noise gradually decreases. According to the Laplace function curve in Figure 1, the probability of outlier output decreases, and the classification accuracy of the DiffPRF_linear algorithm fluctuates occasionally, but it still tends to grow slowly.

By observing Figure 3(d) and 4(d), it is found that when $q = q_4$, the classification accuracy of the DiffPRF_linear algorithm is generally lower than that of Figs. 3(a) ~ 3(c) and Figs. 4(a) ~ 4(c) under the combination of the same q and different d values. This is because the total privacy budget ε_T of each subtree is fixed. In light of the linear allocation strategy, the privacy budget allocated to each layer increases linearly compared to the upper layer. When the value of the coefficient is too large, the privacy budget distributed to the first layer of the tree will be extremely small. The $q_4 = 2$ selected in Figure 3(d) and Figure 4(d) causes the added noise to be too large and the data availability to be extremely low, so the classification accuracy is low. However, with the increase in ε_T, the privacy budget distributed to each decision tree increases, so that the privacy budget distributed to the first layer also increases, the noise level decreases, and the availability of data improves. Therefore, the classification accuracy of the DiffPRF_linear algorithm is gradually improving.

In addition, $d = d_1, d_2$, $q = q_1, q_2$, $\varepsilon_T = 1$, $h = 5,6,7,8,9,10$ are selected to compare the classification accuracy of the DiffPRF_linear algorithm with the uniform, arithmetic, geometric privacy budget allocation method under different decision tree depths. According to Table 1 and Table 2, draw line charts, as shown in Figures 5 and 6. Among them, the coefficient values selected

Figure 3. The classification accuracy of DiffPRF_linear in the Adult dataset. (a) $q_1 = 1.1$ is combined with d_1, d_2 and d_3, (b) $q_2 = 1.2$ is combined with d_1, d_2 and d_3, (c) $q_3 = 1.3$ is combined with d_1, d_2 and d_3, (d) $q_4 = 2$ is combined with d_1, d_2 and d_3

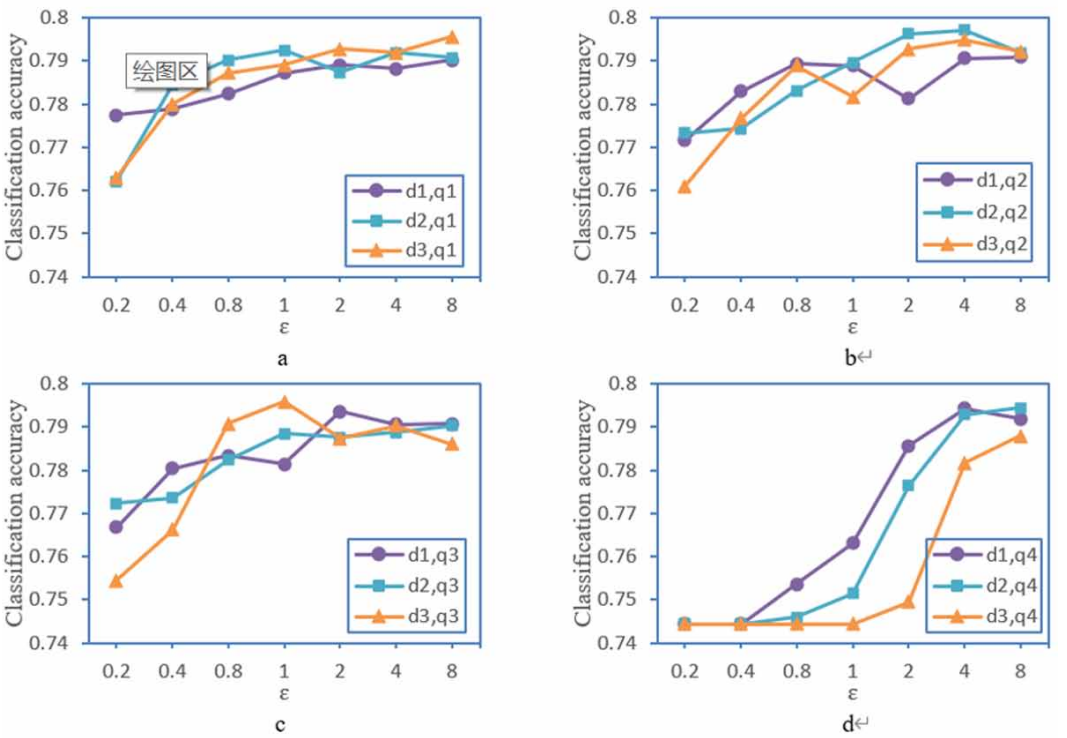
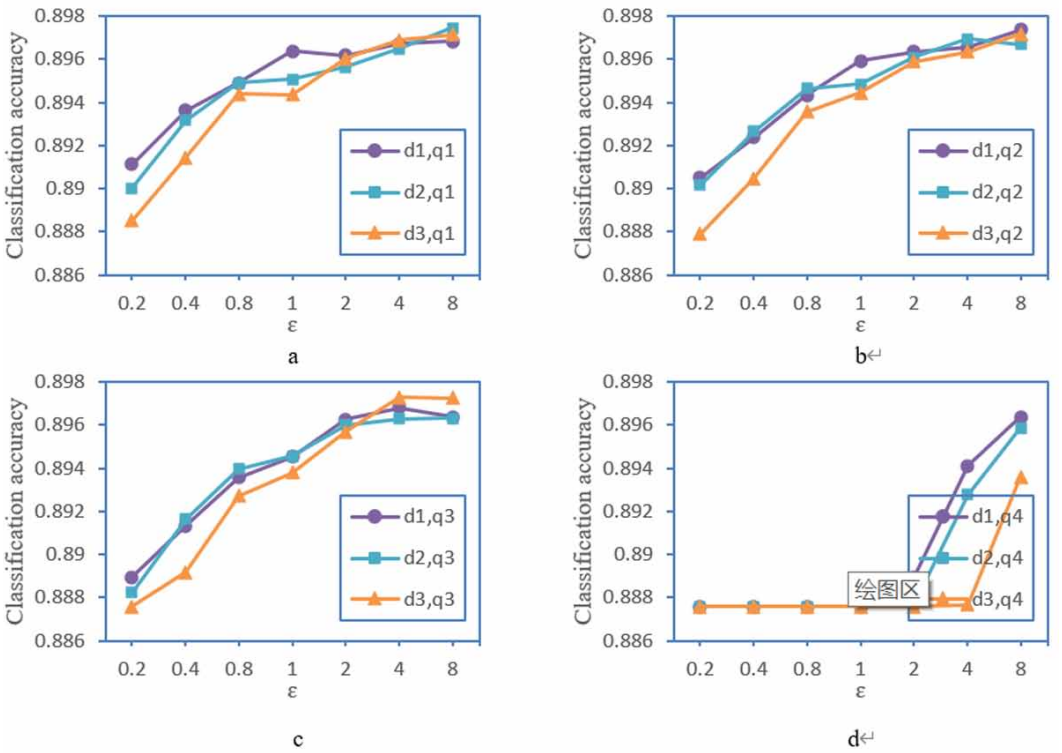


Figure 4. The classification accuracy of DiffPRF_linear in the Bank Marketing dataset. (a) $q_1 = 1.1$ **is combined with** d_1, d_2 **and** d_3, **(b)** $q_2 = 1.2$ **is combined with** d_1, d_2 **and** d_3, **(c)** $q_3 = 1.3$ **is combined with** d_1, d_2 **and** d_3, **(d)** $q_4 = 2$ **is combined with** d_1, d_2 **and** d_3



by the arithmetic distribution are $d = d_2$ and $q = 1$, and the coefficient values selected by the geometric distribution are $d = 0$ and $q = q_2$. $d = d_1$ and $q = q_2$ are selected to be the coefficients of DiffPRF_linear1, $d = d_2$ and $q = q_2$ are selected to be the coefficients of DiffPRF_linear2, $d = d_1$ and $q = q_1$ are selected to be the coefficients of DiffPRF_linear3, and $d = d_2$ and $q = q_1$ are selected to be the coefficients of DiffPRF_linear4. As seen from Figures 5 and 6, due to the strong combinativity of the linear privacy budget allocation strategy, the results of classification accuracy become more diverse, i.e., there are multiple combinations of results. From Figure 5, under h=5,6,7 and 9, the linear privacy budget allocation method in the Adult dataset can achieve better classification results than uniform, arithmetic, and geometric distribution methods. From Figure 6, under h=5,6,7,8 and 9, the linear privacy budget allocation method in the Bank Marketing dataset can achieve better classification results than uniform, arithmetic, and geometric distribution methods. However, when the tree depth is too large, the classification accuracy of the DiffPRF_linear algorithm decreases when the depth of the decision tree increases. This is because the depth of the tree is too large, which causes the privacy budget allocated to each layer of the decision tree to be less. The lower budget is, the larger the added noise, so the classification accuracy will drop significantly.

Table 3 summarizes the characteristics of different methods under differential privacy. In conclusion, the DiffPRF_linear algorithm is more flexible and has a variety of combinations. By changing the values of d and q, in addition to achieving the effect of a uniform, arithmetic and

Table 1. Classification accuracy of different tree depths in the Adult datasets

	h=5	h=6	h=7	h=8	h=9	h=10
RFDPP-Gini	0.772225	0.777981	0.786331	0.790475	0.79176	0.787967
DiffPRF- arithmetic	0.770504	0.777436	0.786403	0.790378	0.788282	0.787979
DiffPRF- geometric	0.769377	0.777193	0.789881	0.787227	0.786973	0.790002
DiffPRF_linear1	0.775485	0.777339	0.79205	0.786391	0.788827	0.786888
DiffPRF_linear2	0.769292	0.779423	0.789554	0.786755	0.789615	0.786428
DiffPRF_linear3	0.769971	0.779569	0.791844	0.786755	0.787227	0.785143
DiffPRF_linear4	0.77048	0.780271	0.782804	0.789227	0.792475	0.788888

Figure 5. Comparison of different tree depths under the Adult dataset

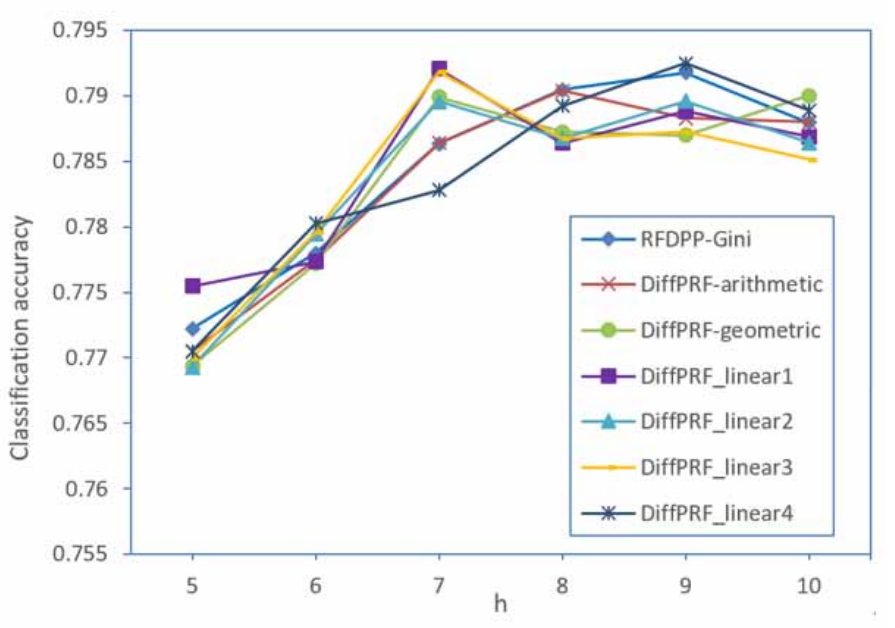


Table 2. Classification accuracy of different tree depths in Bank Marketing datasets

	h=5	h=6	h=7	h=8	h=9	h=10
RFDPP-Gini	0.893329	0.895115	0.895584	0.895449	0.895692	0.895412
DiffPRF- arithmetic	0.893469	0.895228	0.895811	0.895395	0.895768	0.895746
DiffPRF- geometric	0.893858	0.895406	0.89484	0.894904	0.896048	0.895433
DiffPRF_linear1	0.89396	0.895066	0.895212	0.895838	0.895919	0.89546
DiffPRF_linear2	0.892935	0.895293	0.895994	0.895293	0.89485	0.894343
DiffPRF_linear3	0.893734	0.894807	0.895271	0.89518	0.896377	0.894635
DiffPRF_linear4	0.893766	0.895579	0.89566	0.895314	0.895055	0.895363

Figure 6. Comparison of different tree depths under the Bank Marketing dataset

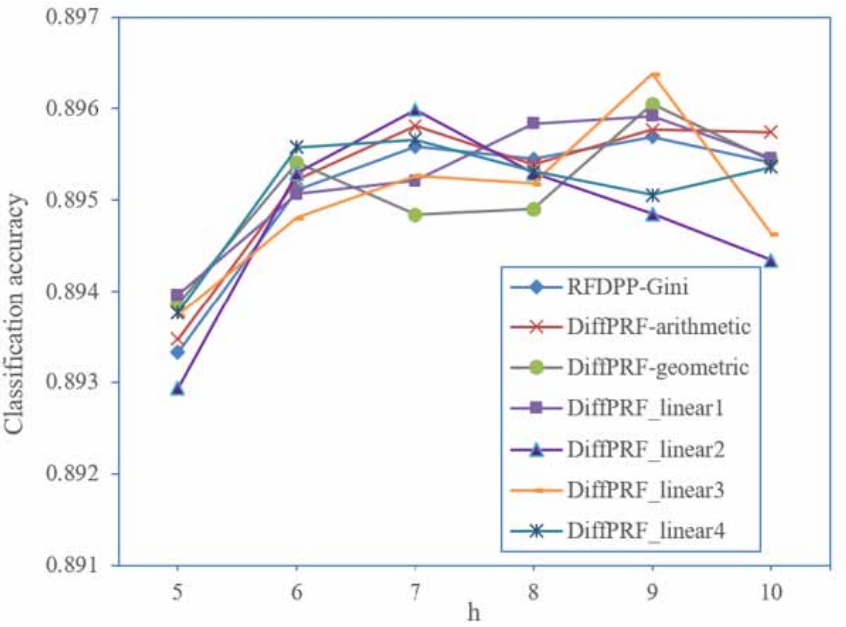


Table 3. Comparison of different methods under differential privacy

	Allocation of privacy budget	**characteristics**
RFDPP-Gini	Assign the same privacy budget to each level of the subdecision tree	Exponential mechanism selection efficiency is low, vulnerable to the influence of tree fan out
DiffPRF-arithmetic	Starting from the root node level, the privacy budget allocated to each subdecision tree level increases with constant d ($d \geq 0$)	The setting of d value is limited by the tree depth h , and the classification effect is relatively general
DiffPRF-geometric	Starting from the level where the root node is located, the privacy budget allocated to each subdecision tree layer is a constant q ($q \geq 1$) times of the previous layer	When ε is constant, a large q value results in a very small privacy budget allocated to the first layer of the decision tree and poor data availability
DiffPRF_linear	Starting from the level where the root node is located, the privacy budget allocated to each level of the subdecision tree is a linear combination of constant d and coefficient q	More flexibility, more combinations to choose from

geometric privacy budget allocation strategy, a better classification effect can also be achieved by combining the coefficient q and the constant term d . It provides more solutions for users to choose the appropriate privacy budget.

CONCLUSION

This paper puts forward a random forest algorithm based on linear privacy budget allocation: DiffPRF_linear. Through the linear privacy budget allocation strategy, users can realize the needs of personalized privacy budget allocation, and provide more schemes for users to reasonably choose

the privacy budget. It can meet the requirements of uniform, arithmetic and geometric of privacy budget allocation at the same time and can be combined with a random forest algorithm to classify data and protect privacy. The results of the experiment indicate that the DiffPRF_linear algorithm has good privacy protection performance and classification accuracy. Of course, when the coefficient q is too large, the privacy budget allocated to each layer is too small, which makes the added noise too large, resulting in poor classification results when ε is small. In the next step, the algorithm will be further optimized, and the linear privacy budget allocation strategy will be combined with other machine learning algorithms to obtain a better classification effect.

ACKNOWLEDGMENT

This research was supported by the National Natural Science Foundation of China [grant number U20A20179], Beijing Key Laboratory of Security and Privacy in Intelligent Transportation, Hebei Key Laboratory of Data Science and Application and Tangshan Key Laboratory of Data Science.

REFERENCES

Breiman, L. (1999). Random forests—Random features. *Machine Learning*, *36*(1/2), 85–103. Advance online publication. doi:10.1023/A:1007563306331

Clifton, C., Kantarcioglu, M., Vaidya, J., Lin, X., & Zhu, M. Y. (2002). Tools for privacy preserving distributed data mining. *SIGKDD Explorations*, *4*(2), 28–34. doi:10.1145/772862.772867

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2001). Introduction to algorithms. *The Journal of the Operational Research Society*, *42*(9), 816. Advance online publication. doi:10.2307/2583667

Cormode, G., Procopiuc, C., Srivastava, D., Shen, E., & Yu, T. (2012). Differentially private spatial decompositions. *2012 IEEE 28th International Conference on Data Engineering*, 20-31. doi:10.1109/ICDE.2012.16

Dumbill & Edd. (2013). A revolution that will transform how we live, work, and think: an interview with the authors of big data. *Big Data, 1*(2), 73. 10.1089/big.2013.0016

Dwork, C. (2006). Differential Privacy. In *Proceedings of the 33rd international conference on Automata, Languages and Programming - Volume Part II (ICALP'06).* Springer-Verlag. doi:10.1007/11787006_1

Dwork, C. (2008). Differential Privacy: A Survey of Results. In *International Conference on Theory and Applications of Models of Computation*. Lecture Notes in Computer Science, vol 4978. Springer. doi:10.1007/978-3-540-79228-4_1

Dwork, C., & Lei, J. (2009). Differential privacy and robust statistics. *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC*, 371–380. doi:10.1145/1536414.1536466

Dwork, C., McSherry, F., Nissim, K., & Smith, A. (2012). Calibrating noise to sensitivity in private data analysis. *Lecture Notes in Computer Science*, *3876*(8), 265–284. doi:10.1007/11681878_14

Dwork, C., Naor, M., Reingold, O., Rothblum, G. N., & Vadhan, S. P. (2009). On the complexity of differentially private data release: Efficient algorithms and hardness results. *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC*, 381–390. doi:10.1145/1536414.1536467

Fearn & Tom. (2006). Classification and regression trees (cart). *Journal of Near Infrared Spectroscopy*, *17*(1), 13. .10.1255/nirn.917

Ishai, Y., Kushilevitz, E., Ostrovsky, R., & Sahai, A. (2006). Cryptography from Anonymity. *Proceedings of the 47th Annual IEEE Symp on Foundations of Computer Science. (FOCS'06)*, 239-248. doi:10.1109/FOCS.2006.25

Jiang, W., & Clifton, C. (2006). A secure distributed framework for achieving k-anonymity. *The VLDB Journal*, *15*(4), 316–333. doi:10.1007/s00778-006-0008-z

Li, M., Quan, D., & Yu, L. (2017). Differentially private real-time data release based on the moving average strategy. *Tehnicki Vjesnik Technical Gazette*, *24*(4). Advance online publication. doi:10.17559/TV-20170415165405

Li, N., Li, T., & Venkatasubramanian, S. (2007). t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. *2007 IEEE 23rd International Conference on Data Engineering*, 106-115. doi:10.1109/ICDE.2007.367856

Li, Y., & Chen, X., Liu, L., An Y., & Li Z. (2020). Random Forest Algorithm for Differential Privacy Protection. *Computer Engineering*, (1), 93–101. doi:10.19678/j.issn.1000-3428.0053592

Li, Y., Wen, W., & Xie, G. (2012). A Survey of research on differential privacy. *Application Research of Computers*, (9), 3201-3205+3211.doi: CNKI:SUN:JSYJ.0.2012-09-003.

Machanavajjhala, A., Kifer, D., Gehrke, J., & Venkitasubramaniam, M. (2007). L-diversity: Privacy beyond k-anonymity. *ACM Transactions on Knowledge Discovery from Data*, *1*(1), 3. doi:10.1145/1217299.1217302

Mcsherry, F., & Talwar, K. (2007). Mechanism design via differential privacy. *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, 94-103. doi:10.1109/FOCS.2007.66

Meng X., & Ci X. (2013). Big data management: Concepts, techniques and challenges. *Journal of Computer Research and Development,* (1), 146-169. doi: CNKI:SUN:JFYZ.0.2013-01-020.

Mu R., Ding L., Song Y., & Lu G. (2016). DiffPRFs: random forest under differential privacy. *Journal on Communications*, (9),175-182. doi: CNKI:SUN:TXXB.0.2016-09-018.

Nissim, K., Mcsherry, F. D., Dwork, C., & Blum, A. L. (2005). Practical privacy: the SuLQ framework. *Proceedings of the Twenty-fourth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, 128–138. doi:10.1145/1065167.1065184

Pan, K., Gong, M., Feng, K., & Wang, K. (2021). Differentially private regression analysis with dynamic privacy allocation. *Knowledge-Based Systems*, *217*(1), 106795. doi:10.1016/j.knosys.2021.106795

Patil, A., & Singh, S. (2014). Differential private random forest. *2014 International Conference on Advances in Computing*, *Communication Information*, *2014*, 2623–2630. doi:10.1109/ICACCI.2014.6968348

Rothe, J. (2002). Some facets of complexity theory and cryptography: A five-lectures tutorial. *ACM Computing Surveys*, *34*(4), 504–549. doi:10.1145/592642.592646

Siau, K., & Wang, W. (2020). Artificial intelligence (ai) ethics: Ethics of ai and ethical ai. *Journal of Database Management*, *31*(2), 74–87. doi:10.4018/JDM.2020040105

Statistics, L. B., & Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5–32. doi:10.1023/A:1010933404324

Sweeney, L. (2002). Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-based Systems*, *10*(5), 571–588. doi:10.1142/S021848850200165X

Tao X., Hu X., & Liu Y. (2013). Overview of big data research. *Journal of System Simulation*, (S1), 142-146. . doi: 10.16182/j.cnki.joss.2013.s1.074.10.16182/j.cnki.joss.2013.s1.074

Turesson, H. K., Kim, H., Laskowski, M., & Roatis, A. (2021). Privacy Preserving Data Mining as Proof of Useful Work: Exploring an AI/Blockchain Design. *Journal of Database Management*, *32*(1), 69–85. doi:10.4018/JDM.2021010104

Wang, Q., Zhang, Y., Lu, X., Wang, Z., Qin, Z., & Ren, K. (2016). Real-time and spatio-temporal crowd-sourced social network data publishing with differential privacy. *IEEE Transactions on Dependable and Secure Computing*, 1–1. doi:10.1109/TDSC.2016.2599873

Wang, T., Wang, Y., & Yen, J. (2019). It's Not My Fault: The Transfer of Information Security Breach Information. *Journal of Database Management*, *30*(3), 18–37. doi:10.4018/JDM.2019070102

Wang, W., & Siau, K. (2019). Artificial intelligence, machine learning, automation, robotics, future of work and future of humanity: A review and research agenda. *Journal of Database Management*, *30*(1), 61–79. doi:10.4018/JDM.2019010104

Wang, X., Han, H., Zhang, Z., & Zheng, X. (2018). Differential privacy budget allocation method for data of tree index. *Jisuanji Yingyong*, *38*(07), 1960–1966.doi: 10.11772 /j.issn.1001-9081.2018010014

Wang Y., Jin X., & Cheng X. (2013). Network big data: present and future. *Chinese Journal of Computers*, (6), 1125-1138. doi: CNKI:SUN:JSJX.0.2013-06-001.

Xiao, X., & Tao, Y. (2007). M-invariance: towards privacy preserving re-publication of dynamic datasets. *Proc of the 2007 ACM SIGMOD Int Conf on Management of Data*, 689-700. doi:10.1145/1247480.1247556

Xiong, P., Zhu, T., & Wang, X. (2014). A Survey on Differential Privacy and Applications. *Chinese Journal of Computers*, *37*(01), 101–122. doi:10.3724/SP.J.1016.2014.00101

Xuan, W. (2019). *Optimization and application of privacy budget in differential privacy protection.* Nanjing University of Posts and Telecommunications. https://kns.cnki.net/KCMS/detail/detail.aspx?dbname=CMFD202001&filename=1019691053.nh

Yang, G. M., Gong, C., Fang, X. J., Ge, B., & Su, S. Z. (2019). Local differential privacy protection for frequent sequence mining. *Journal of Harbin Engineering University*, (11), 1903–1910. doi:10.11990/jheu.201812051

Yanling Dong is a Master Candidate in the College of Science at North China University of Science and Technology. Her main research interests include data security and privacy protection.

Shu fen Zhang is a professor and master supervisor in the College of Science at North China University of Science and Technology. Her main research interests include Cloud computing, intelligent information processing, data security and privacy protection.

Jingcheng Xu is a Master Candidate in College of Science at North China University of Science and Technology. His main research interests include data security and privacy protection.

Haoshi Wang is a Master Candidate in College of Science at North China University of Science and Technology. His main research interests include data security and privacy protection.

Jiqiang Liu received his B.S. (1994) and Ph.D. (1999) degree from Beijing Normal University. He is currently a professor at the School of Computer and Information Technology, Beijing Jiaotong University. He has published over 100 scientific papers in various journals and international conferences. His main research interests are trusted computing, cryptographic protocols, privacy-preserving, and network security.