SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

May 2021, Vol. 64 152101:1–152101:18
https://doi.org/10.1007/s11432-020-2910-1

Differential identifiability clustering algorithms for big data analysis

Tao SHANG[1*], Zheng ZHAO[2], Xujie REN[1] & Jianwei LIU[1]

[1]*School of Cyber Science and Technology, Beihang University, Beijing* 100083*, China;*
[2]*School of Electronic and Information Engineering, Beihang University, Beijing* 100083*, China*

Received 18 January 2020/Accepted 18 April 2020/Published online 31 March 2021

Abstract Individual privacy preservation has become an important issue with the development of big data technology. The definition of ρ-differential identifiability (DI) precisely matches the legal definitions of privacy, which can provide an easy parameterization approach for practitioners so that they can set privacy parameters based on the privacy concept of individual identifiability. However, differential identifiability is currently only applied to some simple queries and achieved by Laplace mechanism, which cannot satisfy complex privacy preservation issues in big data analysis. In this paper, we propose a new exponential mechanism and composition properties of differential identifiability, and then apply differential identifiability to k-means and k-prototypes algorithms on MapReduce framework. DI k-means algorithm uses the usual Laplace mechanism and composition properties for numerical databases, while DI k-prototypes algorithm uses the new exponential mechanism and composition properties for mixed databases. The experimental results show that both DI k-means and DI k-prototypes algorithms satisfy differential identifiability.

Keywords differential identifiability, differential privacy, k-means, k-prototypes, big data

Citation Shang T, Zhao Z, Ren X J, et al. Differential identifiability clustering algorithms for big data analysis. Sci China Inf Sci, 2021, 64(5): 152101, https://doi.org/10.1007/s11432-020-2910-1

1 Introduction

With the development of information society, big data has become an important strategic resource which can be used to obtain valuable information from data analysis. The analysis of big data has shown to be beneficial to many fields such as medical, banking, and commerce. However, individual privacy preservation has become one of the biggest issues with the development of big data technology. Privacy preservation notions have achieved many advances along with the continuous growth of collection and analysis of big data. Many privacy definitions and applications for releasing data securely have been introduced in literatures (see [1, 2] for surveys). The most widely accepted notion is differential privacy (DP) developed by a series of studies [3–7]. The basic idea is that any individual in a database has only a limited influence on the output of the database to hide the contribution of any single individual. Dwork et al. [6, 7] also introduced two types of differential privacy, namely unbounded differential privacy and bounded differential privacy [8], depending on which two databases are considered to be neighbors.

Because privacy is a social notion with many facets, research fields examine various facets of privacy to understand strength and weakness. In 2011, Kifer and Machanavajjhala [8] argued that differential privacy is not robust to arbitrary background knowledge and cannot provide privacy and utility without the assumption of data. So differential privacy is appropriate only when data records are independent. Cormode [9] proposed that differential privacy does not prevent inferential disclosure. It is possible to conclude potential privacy information on an individual by non-trivial accuracy. In 2012, Lee and Clifton [10] argued that the definition of differential privacy has an obvious flaw. Privacy parameter ε is an index for evaluating security, but it only limits the difference in the query results of two neighboring databases. In other words, ε limits how much information is revealed from the released statistic, but it is

* Corresponding author (email: shangtao@buaa.edu.cn)

difficult to directly relate it to the identification probability of individuals. The definition does not meet the legal requirements of privacy, which is required to protect individually identifiable data. As a result, they proposed differential identifiability (DI) that provides the same privacy guarantee as differential privacy, but the privacy parameter ρ limits the probability estimate that an individual is re-identified. They showed that a query function f is a simple aggregate query, similar to the Laplace mechanism of differential privacy, and the Laplace mechanism can be applied to achieve differential identifiability. So the implementation mechanism can be regarded as equivalent; i.e., differential identifiability can provide privacy preservation similar to differential privacy. Thus it has practical application in privacy preservation of data analysis.

By the similarity of noise ratio, it is evident that ρ-differential identifiability and ε-differential privacy are comparable in terms of privacy. Achieving this privacy is not an easy problem by means of setting ε. However, ρ-differential identifiability can provide an easy parameterization approach for practitioners so that they can easily choose the privacy parameter ρ to match the legal definitions of privacy. At present, the practical applicability of differential privacy is not limited to simple query functions. It is widely applied to data mining. Blum et al. [5] first proposed k-means algorithm combined with differential privacy and Dwork [11] perfected the work of setting privacy parameter ε. Then a series of related studies have been developed. For example, Li et al. [12] proposed a data publishing method ICMD-DP that performs differential privacy on the results of insensitive clustering algorithm for mixed data (ICMD), and Zhao et al. [13] proposed the optimized Canopy to improve initial centers of differential privacy k-means algorithm, which is based on MapReduce framework. Bugata and Drotar [14] defined the objective function whose maximization is equivalent to minimum redundancy maximum relevance (mRMR) and provided the generalization of the mRMR method. Lee and Clifton [10] only studied differential identifiability for each dimension attribute in an input database, not for the whole input database. For data mining, setting privacy parameter ρ and analyzing m (the number of possible worlds) to achieve differential identifiability are more complex. It is necessary to study the security and utility of differential identifiability data mining algorithms, and compare it with the results of differential privacy.

In order to explore differential identifiability for big data mining, we address a new mechanism and composition properties of differential identifiability with reference to differential privacy. In this paper, we show that the exponential mechanism of differential privacy can also be used to satisfy differential identifiability, which can make differential identifiability applicable to the situation where an input database has non-numerical attributes. We also develop sequential composition and parallel composition of differential identifiability for complex privacy preservation issues. Based on the exponential mechanism and composition properties, differential identifiability, as with differential privacy, can also be combined with data mining algorithms.

The main contributions of this paper are summarized as follows.

(1) Exponential mechanism and composition properties of differential identifiability. According to differential privacy, both of them are the basis of differential identifiability in big data mining. Exponential mechanism can be used for the situation where a database is non-numerical. Composition properties can be used for implementing complex privacy preservation by means of multiple privacy preservation mechanisms.

(2) Clustering algorithms with differential identifiability for big data analysis. From the perspective of big data mining, we propose DI k-means and DI k-prototypes algorithms, which can be used for numerical data and mixed data, respectively. These algorithms are given on MapReduce framework by analyzing the selection of privacy parameters ρ and m, and compared with the results of differential privacy algorithms.

This paper is structured as follows. In Section 2, we introduce the related work of privacy preservation. Section 3 provides the preliminaries, including differential identifiability, k-means algorithm, k-prototypes algorithm, and MapReduce framework. Section 4 focuses on exponential mechanism and Section 5 focuses on composition properties of differential identifiability. Then the analysis of DI k-means and DI k-prototypes algorithms are given in Sections 6 and 7. Section 8 is our conclusion.

2 Related work

Most of modern privacy preservation research is carried out in the background of databases, which can be divided into two categories, namely data clustering and theoretical frameworks [2]. Privacy

preservation in the category of data clustering developed from the original k-anonymity to l-diversity and t-closeness. Samarati and Sweeney [15–17] first proposed k-anonymity, which requires that there are at least k individuals in each equivalence class, and the quasi-identifier of each individual is at least the same as that of other $k-1$ individuals. The k-anonymity only deals with quasi-identifiers, but not with sensitive attributes, making it vulnerable to homogeneous attacks and background knowledge attacks. To improve the shortcomings of k-anonymity, Machanavajjhala et al. [18] proposed the l-diversity method to ensure that sensitive attributes in each equivalence class have at least l different values in 2006. l-diversity cannot effectively prevent similarity attacks and might disclose more private information to the attacker in some circumstances. In 2007, Li et al. [19] put forward t-closeness, which requires the probability distribution of sensitive attribute values in the equivalence class and data set is less than the threshold t. t-closeness can effectively prevent similarity attacks. In 2010, they put forward the improved (n,t)-closeness on the basis of the original t-closeness [20]. This method is more flexible and can enhance data utility, but did not provide a specific calculation process.

Different from the category of data clustering, theoretical frameworks include differential privacy and its privacy preservation methods, which can provide powerful preservation for data sets. In 2006, Dwork [7] proposed differential privacy which requires the outputs of two neighbor databases are close, so that the information gain to the attacker is limited. Dwork et al. [6,7] introduced two types of differential privacy, namely bounded differential privacy and unbounded differential privacy. In bounded differential privacy, if one data set is different from another data set by replacing an individual, the two data sets are considered to be neighbor data sets. In unbounded differential privacy, if one data set is different from another data set by adding or deleting an individual, the two data sets are considered to be adjacent data sets. In 2013, Lee and Clifton [10] argued that the parameter of differential privacy ε limits how much information is revealed from the released statistic, but it is difficult to directly relate it to identification probability of individuals. So they proposed the definition of ρ-differential identifiability. It provides the same privacy guarantee as differential privacy, but the privacy parameter ρ limits the probability estimate that an individual is re-identified. In 2013, Li et al. [21] analyzed the advantages and disadvantages of differential privacy and differential identifiability, proposed a membership privacy framework, and transformed the parameters of privacy preservation concept into a series of distribution clusters, providing theoretical support for the development of new privacy concepts. Under this framework, researchers can achieve better differential privacy and differential identifiability by limiting some distribution conditions. In 2016, Backes et al. [22] took advantage of the concept of membership privacy framework and obtained a weak differential privacy preservation by limiting some distribution conditions to produce a small amount of noise under the same circumstances.

Aggarwal et al. [23] first proposed the method of clustering anonymization to achieve privacy preservation. The basic idea is to conduct clustering operation on the original data set and then replace all records with the cluster center and publish them together with the cluster features. Because most of data are severely damaged by this method, the utility of data is greatly reduced. Byun et al. [24] also proposed a method to achieve anonymity based on clustering technology. This method adopts multiple clustering methods without considering the influence of outliers on clustering results, and the data availability is poor. These two methods cannot effectively protect privacy. In 2005, Blum et al. [5] first proposed k-means algorithm combined with differential privacy. In the process of calculating the cluster center, only a small amount of noise needs to be added to the query results of sum function and counting function to ensure the security of the cluster center. This algorithm has a high sensitivity and does not give a way to set the privacy budget. In 2007, Nissim et al. [25] proposed pk-means algorithm to make the final result of clustering meet the definition of differential privacy. The specific process of how to calculate the error lower bound and function sensitivity was given. In 2014, Jafer et al. [26] proposed the TOP_Diff algorithm, which can reduce the operation time of anonymization of large data sets by k-anonymity and differential privacy after feature extraction of original data sets, and obtain higher data utility by mutually adjusting the k value of the anonymization level and the privacy budget. The algorithm can only be applied in non-numerical data. In 2017, Li et al. [12] proposed a data preservation method of differential privacy for mixed attribute data. ICMD clustering algorithm was used to anonymize the data set. For the non-numerical data and numerical data, different methods were used to calculate the cluster center. A total order function was introduced to ensure that the anonymized data set meets differential privacy. The algorithm differentiates the query sensitivity from single data to equivalent class through clustering, reduces the risk of information loss and privacy disclosure, increases data utility, and has better data protection effect than standard differential privacy.

3 Preliminaries

3.1 Differential identifiability

A database D can be considered as a finite multiset. Each attribute value is a fixed value in the universe U. Each entry in U can correspond to an individual in the database the privacy of which should be protected. $I(t)$ denotes the identity of the individual corresponding to the entry t in U. $\mathcal{I}_D = \{I(t)|t \in D\}$ denotes the set of individuals which belong to D. $D' \subset D$ is a database owning one less individual than D, i.e., $|D'| = |D| - 1$.

Lee and Clifton [10] argued that differential privacy parameter ε limits how much one individual can affect an output, not how much information can be revealed about an individual. This does not match the legal definitions of privacy, which requires to protect individually identifiable data. Thus they proposed the definition of ρ-differential identifiability that can provide the same guarantees as differential privacy, but ρ limits the probability estimate that an individual belongs to the input database. The definition is as follows.

Definition 1 (ρ-differential identifiability [10])**.** A randomized mechanism M is said to satisfy ρ-differential identifiability if for all databases D, any $D' = D - t^*$, for any entry $t \in U - D'$:

$$\Pr[I(t) \in \mathcal{I}_D | M(D) = R, D'] \leqslant \rho. \tag{1}$$

The definition of ρ-differential identifiability limits the identifiable risk of any individual in the universe U, and thus the posterior probability that any individual t belongs to the database is less than or equal to ρ after an adversary observes the output response R. In order to calculate the posterior probability, it is necessary to assume prior beliefs that the adversary may have.

To measure the adversary's confidence in making an inference, the proposed definition assumes that there exists a possible world model in which the adversary considers the set of all possible databases. Every possible world consists of D' and one entry in U. Given the adversary's prior knowledge $\mathcal{L} = \langle U, D', \mathcal{I}_{D'} \rangle$, the set of all possible databases Ψ is

$$\Psi = \{D' \cup \{t\} | t \in U \wedge t \notin D'\}. \tag{2}$$

Every possible world $\omega \in \Psi$ is equally likely to be D. Only one of the databases in Ψ is the true database which generates the output response R. In other words, only one individual is uncertain, and this individual must be drawn uniform from $m = |\Psi| = |U| - |D'|$ possible individuals with the probability between 0 and 1. At the same time, Lee and Clifton [10] experimentally proved that when the value of ρ is close to the correct probability of a random guess, the output response is barely useful and the privacy goal is also violated. Thus ρ-differential identifiability will be useful when $\rho > \frac{1}{m}$.

They also showed how much noise should be added to satisfy differential identifiability, given the sensitive range of a query function f.

Theorem 1 ([10])**.** For a randomized mechanism M and an arbitrary adversary, if $\lambda = \frac{\Delta f}{\ln \frac{(m-1)\rho}{1-\rho}}$, M satisfies ρ-differential identifiability where $m = |\Psi|$ and Δf is the sensitive range of f.

As with the Laplace mechanism developed for differential privacy [6], random noise Y is added to every query result $R = f(D) + Y$, where Y is a random variable drawn from Laplace distribution, i.e., $Y \sim \mathrm{Lap}(\lambda)$. Differential identifiability can be also achieved by Laplace mechanism.

Differential identifiability (as with differential privacy) can provide strong privacy preservation. Under the equal likeliness between possible worlds, there exists a one-to-one mapping relationship between the privacy parameters of two definitions. Based on the proof of Laplace mechanism, it is reasonable to believe that analogous mechanism and properties to differential privacy can be developed to achieve differential identifiability. If differential identifiability is applied to data mining algorithms, these mechanism and properties should be developed to provide an easy parameterization approach for practitioners such as the following.

(1) Differential identifiability mechanism for non-numerical data. According to differential privacy, random noise drawn from Laplace distribution only can be added to numerical data. In fact, there is much non-numerical data required for privacy preservation.

(2) Composition of differential identifiability. A complex privacy preservation issue generally requires to use multiple privacy preservation mechanisms. So we should know what level of privacy preservation differential identifiability can provide under composition.

(3) Algorithms with differential identifiability. Based on (1) and (2), differential identifiability, as with differential privacy, can be used to protect the results of complex privacy preservation queries. Furthermore, we can combine differential identifiability with data mining algorithms.

3.2 Clustering algorithms

(1) k-means algorithm. k-means is a classical clustering algorithm that is the most widely used in data mining. The basic idea is to divide data records in a database into the same cluster. k-means randomly assigns k initial centers, divides each data record into the nearest center, and iterates repeatedly until k-means algorithm converges. The basic flow of k-means algorithm is described as follows.

Step1. Randomly assign k initial centers $u_1, u_2, \ldots, u_k$;

Step2. Repeat: divide each data record into the nearest center and recalculate the k cluster centers;

Step3. Until: cluster centers converges.

Assuming a database D with n data records $x_1, x_2, \ldots, x_n$ and d dimensions, Euclidean metric is generally used to measure the similarity between data records and cluster centers.

$$\mathrm{dist}(x_i, u_j) = \sqrt{\sum_{l=1}^{d}(x_{il} - u_{jl})^2}. \tag{3}$$

The sum of the squared error (SSE) is used as the objective function of k-means algorithm, and it is also the index used to measure different clustering results.

$$\mathrm{SSE} = \sum_{i=1}^{k}\sum_{x\in U_i}(\mathrm{dist}(x, u_i))^2. \tag{4}$$

Eq. (4) denotes the sum of squares of the distance between data records x in cluster U_i and the center u_i of cluster U_i. The optimal clustering result should make SSE have the minimum.

(2) k-prototypes algorithm. Huang [27] proposed a k-prototypes algorithm which is based on k-means and k-modes algorithms. The algorithm introduces a parameter to control the weight of categorical attributes for clusters. k-prototypes is a classical partitional clustering algorithm for mixed attributes.

Assuming a database D with n data records $x_1, x_2, \ldots, x_n$, $x_i = [x_{i1}, x_{i2}, \ldots, x_{id}]$ has d dimensions. k-prototypes algorithm divides the database D into k different clusters by minimizing a specific objective function. The objective function (also called cost function) is

$$E = \sum_{j=1}^{k}\sum_{i=1}^{n} y_{ij} d(x_i, u_j), \tag{5}$$

where $u_j = [u_{j1}, u_{j2}, \ldots, u_{jd}]$ is the representative vector or prototype for cluster U_j, i.e., the cluster center, and y_{ij} is an element of a partition matrix $Y_{n\times k}$. $d(x_i, u_j)$ is a similarity measure of mixed attributes. k-prototypes introduces the similarity measure as

$$d(x_i, u_j) = \sum_{l=1}^{d_r}(x_{il}^r - u_{jl}^r)^2 + \gamma_j \sum_{l=1}^{d_c}\delta(x_{il}^c, u_{jl}^c), \tag{6}$$

where

$$\delta(x_{il}^c, u_{jl}^c) = \begin{cases} 0, & x_{il}^c = u_{jl}^c, \\ 1, & \text{otherwise}, \end{cases}$$

x_{il}^r and u_{jl}^r are the values of numerical attributes, whereas x_{il}^c and u_{jl}^c are the values of categorical attributes for data record x_i and the prototype of cluster U_j. d_r and d_c are the number of numeric and categorical attributes. γ_j is the weight of categorical attributes for clusters.

The basic flow of k-prototypes algorithm is described as follows.

Step1. Randomly assign k initial centers $u_1, u_2, \ldots, u_k$ in a database D.

Step2. Repeat: divide each data record into the nearest center by (6) and update the k cluster centers. For numerical attributes, calculate the means of data records. For categorical attributes, select the data record with highest frequency in each cluster.

Step3. Until: cost function E converges.

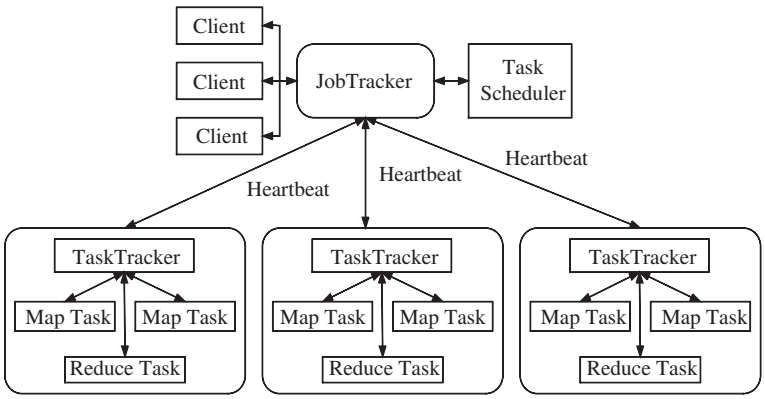

Figure 1 MapReduce architecture.

3.3 MapReduce framework

As a prevailing big data platform, Hadoop provides a distributed computing environment with open-source scalability and high reliability. It contains the components of HDFS and MapReduce [28]. HDFS provides distributed storage services and MapReduce provides parallel computing services for big data based on distributed file systems, using the idea of "divide and rule".

As shown in Figure 1, the components of MapReduce include Client, JobTracker, TaskTracker and Task. Client submits user-written programs to JobTracker, and the users can view the job status through the interface provided by Client. JobTracker is responsible for resource monitoring and job scheduling. JobTracker monitors the status of TaskTracker and jobs. TaskTracker periodically reports resource usage and task progress to JobTracker, receives commands sent by JobTracker, and performs corresponding operations. Task is composed of Map Task and Reduce Task. Map Task and Reduce Task are performed by TaskTracker. The input of Map Task is a number of data chunks divided according to an input database D, and Map Task converts file splits to (key, value) pairs. Reduce Task receives the (key, value) pairs output by Map Task and sorts the (key, value) pairs according to key.

4 Exponential mechanism of differential identifiability

Noise Y is a random variable drawn from Laplace distribution in Laplace mechanism. It only can be added to numerical data. An issue is the design of mechanisms for the situation where an output is non-numerical data. In 2007, McSherry and Talwar [29] proposed a general differential privacy mechanism, called exponential mechanism, which can be applied to non-numerical data to achieve differential privacy. We also expect that the analogous mechanism can be developed for non-numerical data to achieve differential identifiability.

The goal of exponential mechanism is to map an input database D from the universe U to some output r from a range Range and does not provide additional information for an adversary, which achieves differential identifiability.

As with the Laplace mechanism of differential identifiability, a general exponential mechanism of differential identifiability is also decided by a score function sf: $U \times \text{Range} \to \mathbb{R}$. The score function sf generates a real-valued score for each pair (D, r) from $U \times \text{Range}$. The higher score of r means the greater probability of outputting r. Given the input database D, the exponential mechanism aims to return an entity $r \in \text{Range}$ making $\text{sf}(D, r)$ approximately maximized, while the returned entity satisfies differential identifiability.

Definition 2. For any score function sf: $U \times \text{Range} \to \mathbb{R}$, a randomized mechanism M, we define the mechanism M chooses r from Range with probability proportional to $\exp(\text{sf}(D, r) \ln \frac{(m-1)\rho}{1-\rho}/2\Delta\text{sf})$ where $\rho > \frac{1}{m}$,

$$\Pr[M(D) = r : r \in \text{Range}] \propto \exp\left(\frac{\text{sf}(D, r) \ln \frac{(m-1)\rho}{1-\rho}}{2\Delta\text{sf}}\right). \tag{7}$$

According to Definition 2, a small additive change to $\text{sf}(D, r)$, which might be caused by a single entry,

has limited multiplicative influence on the distribution of outputs. Nevertheless, the probability that the mechanism M outputs r increases exponentially with its real-valued score on the input D. The output of the mechanism M obviously biases the distribution towards high-score outputs. Definition 2 can capture any differential identifiability mechanism M by taking $\mathrm{sf}(D,r)$ as the logarithm of the probability that $M(D)$ outputs r. Such transformation does not leak any additional information about the mechanism M. For any score function sf, $\Delta\mathrm{sf}$ is the sensitive range of the score function sf when it is applied to the input database that has one entry unknown.

Proposition 1. For an arbitrary adversary, M is said to satisfy ρ-differential identifiability if Eq. (7) holds.

Proof. The probability of (7) at r is

$$\frac{\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})}{\int\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})\mathrm{d}r}.$$

$\Gamma(t)$ denotes the identifiable risk for an individual t, i.e., the probability which the adversary believes $t\in D$ given the output result.

$$\begin{aligned}\Gamma(t)&=\Pr[I(t)\in\mathcal{I}_D|M(D)=r,D']\\&=\Pr[D=D'\cup\{t\}|M(D)=r,D']\\&=\frac{\Pr[D=D'\cup\{t\}]}{\Pr[M(D)=r]}\cdot\Pr[M(D)=r|D=D'\cup\{t\}]\\&=\frac{\Pr[D=D'\cup\{t\}]\cdot\Pr[M(D'\cup\{t\})=r]}{\sum_{\omega\in\Psi}\Pr[\omega]\Pr[M(\omega)=r]}\\&=\frac{\frac{\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})}{\int\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})\mathrm{d}r}}{\frac{\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})}{\int\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})\mathrm{d}r}+\sum_{\omega\in\Psi,\omega\neq D}\frac{\exp(\frac{\mathrm{sf}(\omega,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})}{\int\exp(\frac{\mathrm{sf}(\omega,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})\mathrm{d}r}}.\end{aligned}$$

Because $|\mathrm{sf}(\omega_1)-\mathrm{sf}(\omega_1)|\leqslant\Delta\mathrm{sf}$, the single individual t can change $\mathrm{sf}(D,r)$ by most $\Delta\mathrm{sf}$. We can get

$$\begin{aligned}\sum_{\omega\in\Psi,\omega\neq D}\frac{\exp(\frac{\mathrm{sf}(\omega,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})}{\int\exp(\frac{\mathrm{sf}(\omega,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})\mathrm{d}r}&\geqslant(m-1)\frac{\exp(\frac{(\mathrm{sf}(D,r)-\Delta\mathrm{sf})\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})}{\int\exp(\frac{(\mathrm{sf}(D,r)+\Delta\mathrm{sf})\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})\mathrm{d}r}\\&=(m-1)\exp\left(-\ln\frac{(m-1)\rho}{1-\rho}\right)\frac{\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})}{\int\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})\mathrm{d}r}.\end{aligned}$$

Then we apply it to $\Gamma(t)$,

$$\begin{aligned}\Gamma(t)&=\frac{\frac{\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})}{\int\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})\mathrm{d}r}}{\frac{\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})}{\int\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})\mathrm{d}r}+\sum_{\omega\in\Psi,\omega\neq D}\frac{\exp(\frac{\mathrm{sf}(\omega,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})}{\int\exp(\frac{\mathrm{sf}(\omega,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})\mathrm{d}r}}\\&\leqslant\frac{\frac{\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})}{\int\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})\mathrm{d}r}}{\frac{\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})}{\int\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})\mathrm{d}r}+(m-1)\mathrm{e}^{-\ln\frac{(m-1)\rho}{1-\rho}}\frac{\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})}{\int\exp(\frac{\mathrm{sf}(D,r)\ln\frac{(m-1)\rho}{1-\rho}}{2\Delta\mathrm{sf}})\mathrm{d}r}}\\&=\frac{1}{1+\frac{(1-\rho)}{\rho}}=\rho.\end{aligned}$$

Table 1 Main notations

Notation	Meaning
U	Universe
$D \subset U$	Database to be queried
$D' = D - i$	Subset of D missing one individual
$t \in U$	Data associated with an individual
$I(t)$	Identity of an individual corresponding to t
$\mathcal{I}_D = \{I(t)\|t \in D\}$	Set of individuals which belong to D
Ψ	All possible databases
f	Query function
Δf	Sensitive range of f
sf	Score function
M	Randomized mechanism
$\Gamma(t)$	Identifiable risk for an individual t

Thus the mechanism M satisfies ρ-differential identifiability:

$$\Gamma(t) = \Pr[I(t) \in \mathcal{I}_D | M(D) = r, D'] \leqslant \rho.$$

And $\rho > \frac{1}{m}$, ρ-differential identifiability makes sense.

The result shows that Definition 2 will be useful for non-numerical data to achieve differential identifiability. Under the assumption that each possible world is equally likely, Definition 2 also shows that ε-differential privacy exponential mechanism can satisfy ρ-differential identifiability when $\varepsilon \leqslant \ln \frac{(m-1)\rho}{1-\rho}$. Differential identifiability exponential mechanism M can choose and output r with the probability proportional to $\exp(\mathrm{sf}(D, r) \ln \frac{(m-1)\rho}{1-\rho}/2\Delta\mathrm{sf})$. Note that the difference between the probabilities of outputting individuals decreases as ρ decreases. The characteristic is similar to exponential mechanism of differential privacy.

For convenience, main notations are listed in Table 1.

5 Composition of differential identifiability

As we know, any privacy preservation mechanism needs to address the issue of composition, because a complex privacy preservation issue generally requires to use multiple privacy preservation mechanisms. However, the properties of ρ-differential identifiability under composition are still not clarified. The mapping relationship between two definitions cannot address the issue. What level of privacy preservation can differential identifiability provide when a given adversary can compose several output responses satisfying differential identifiability or operate structurally disjoint datasets?

For a series of mechanisms with ρ_i-differential identifiability, sequential composition provides $(m^{n-1} \prod_{i=1}^{n} \rho_i)$-differential identifiability. It is normal that privacy level degrades as more information is leaked. As an important strategy, we need to control the privacy parameter in a good way.

When an adversary operates structurally disjoint datasets, parallel composition provides $(\min \rho_i)$-differential identifiability. For example, the parallel clustering algorithm in a distributed system is a typical application.

5.1 Sequential composition

In order to formally define sequential composition, we consider a composition scenario similar to that in [30,31]. In this scenario, an adversary tries to break privacy and figure out whether an individual is in the database by means of analyzing the sequential outputs and adaptively choosing queries to perform. The adversary can arbitrarily control which query to perform, and which mechanism satisfying differential identifiability to be used for each query. The adversary also can make decisions adaptively according to preceding responses.

Proposition 2. Given a set of n mechanisms $M_1, \ldots, M_n$, each $M_i, i \in [1, n]$ provides ρ_i-differential identifiability. For all databases D, the sequential composition $\mathcal{M}(D) = (M_1(D), \ldots, M_n(D))$ provides $(m^{n-1} \prod_{i=1}^{n} \rho_i)$-differential identifiability.

Proof. For any sequence $r_i \in \text{Range}(M_i)$ of outputs, we assume that $\forall i \in [1,n]$, $M_i(D) = r_i$ and $\mathcal{R} = \{r_i | i \in [1,n]\}$. The probability of outputting $\mathcal{R}$ from the sequential composition $\mathcal{M}$ is

$$\Pr[\mathcal{M}(D) = \mathcal{R}] = \prod_{i=1}^{n} \Pr[M_i(D) = r_i]. \tag{8}$$

According to Definition 1, each mechanism M_i satisfies

$$\begin{aligned}\Gamma_i(t) &= \Pr[I(t) \in \mathcal{I}_D | M_i(D) = r_i, D'] \\ &= \frac{\Pr[D = D' \cup \{t\}] \cdot \Pr[M_i(D' \cup \{t\}) = r_i]}{\Pr[M_i(D) = r_i]} \\ &\leqslant \rho_i. \end{aligned} \tag{9}$$

Then we apply the definition of differential identifiability to the sequential composition $\mathcal{M}$,

$$\begin{aligned}\Gamma(t) &= \Pr[I(t) \in \mathcal{I}_D | \mathcal{M}(D) = \mathcal{R}, D'] \\ &= \Pr[D = D' \cup \{t\} | \mathcal{M}(D) = \mathcal{R}, D'] \\ &= \frac{\Pr[D = D' \cup \{t\}]}{\Pr[\mathcal{M}(D) = \mathcal{R}]} \cdot \Pr[\mathcal{M}(D) = \mathcal{R} | D = D' \cup \{t\}] \\ &= \frac{\Pr[D = D' \cup \{t\}] \cdot \Pr[\mathcal{M}(D' \cup \{t\}) = \mathcal{R}]}{\Pr[\mathcal{M}(D) = \mathcal{R}]} \\ &= \frac{\Pr[D = D' \cup \{t\}] \cdot \prod_{i=1}^{n} \Pr[M_i(D' \cup \{t\}) = r_i]}{\prod_{i=1}^{n} \Pr[M_i(D) = r_i]}. \end{aligned} \tag{10}$$

Each mechanism M_i satisfying ρ_i-differential identifiability gives

$$\begin{aligned}\Gamma(t) &= \frac{\Pr[D = D' \cup \{t\}] \cdot \prod_{i=1}^{n} \Pr[M_i(D' \cup \{t\}) = r_i]}{\prod_{i=1}^{n} \Pr[M_i(D) = r_i]} \\ &\leqslant \rho_1 \cdot \frac{\rho_2}{\Pr[D = D' \cup \{t\}]} \cdots \frac{\rho_n}{\Pr[D = D' \cup \{t\}]} \\ &= m^{n-1} \prod_{i=1}^{n} \rho_i. \end{aligned} \tag{11}$$

The belief of the adversary on $I(t) \in \mathcal{I}_D$, i.e., $D = D' \cup \{t\}$ after observing the sequence composition $\mathcal{R}$ of outputs can be bounded by

$$\Pr[I(t) \in \mathcal{I}_D | \mathcal{M}(D) = \mathcal{R}, D'] \leqslant m^{n-1} \prod_{i=1}^{n} \rho_i. \tag{12}$$

The sequential composition $\mathcal{M}(D)$ provides $(m^{n-1} \prod_{i=1}^{n} \rho_i)$-differential identifiability. Inequality holds at $n = 1$ that satisfies ρ_1-differential identifiability. ρ-differential identifiability makes sense when $\rho > \frac{1}{m}$. Thus

$$\prod_{i=1}^{n} \rho_i > \frac{1}{m^n}, \quad m^{n-1} \prod_{i=1}^{n} \rho_i > \frac{1}{m}, \tag{13}$$

$(m^{n-1} \prod_{i=1}^{n} \rho_i)$-differential identifiability makes sense.

Sequential composition is important for any privacy preservation method that needs to address multiple queries. Privacy definitions that are robust under sequential composition can be used for the interactive mode to answer multiple queries, but also require to control the privacy parameter in a good way.

5.2 Parallel composition

In the case that the mechanisms are applied to structurally disjoint datasets, the bound of composition can be further improved. If the input database is divided into disjoint and independent subsets that the

output of each of which is satisfied with differential identifiability, the privacy level provided by parallel composition is the best privacy level of the sequence of mechanisms. The parallel composition scenario is similar to that in [32].

Proposition 3. Given a set of n mechanisms $M_1, \ldots, M_n$, each $M_i, i \in [1, n]$ provides ρ_i-differential identifiability. D_i is the arbitrary disjoint subset of the input database D. The sequence of mechanisms $M_i(D_i)$ provides $(\min \rho_i)$-differential identifiability.

Proof. For any sequence $r_i \in \text{Range}(M_i)$ of outputs, we assume that $\forall i \in [1, n]$, $M_i(D_i) = r_i$ and $\mathcal{R} = \{r_i | i \in [1, n]\}$. The probability of outputting $\mathcal{R}$ from the sequence of $M_i(D_i)$ is

$$\Pr[\mathcal{M}(D) = \mathcal{R}] = \prod_{i=1}^{n} \Pr[M_i(D_i) = r_i]. \tag{14}$$

Then we apply the definition of differential identifiability to each $M_i(D_i)$,

$$\begin{aligned}
\Gamma(t) &= \Pr[I(t) \in \mathcal{I}_D | \mathcal{M}(D) = \mathcal{R}, D'] \\
&= \Pr[D = D' \cup \{t\} | \mathcal{M}(D) = \mathcal{R}, D'] \\
&= \frac{\Pr[D = D' \cup \{t\}]}{\Pr[\mathcal{M}(D) = \mathcal{R}]} \cdot \Pr[\mathcal{M}(D) = \mathcal{R} | D = D' \cup \{t\}] \\
&= \frac{\Pr[D = D' \cup \{t\}] \cdot \Pr[\mathcal{M}(D' \cup \{t\}) = \mathcal{R}]}{\Pr[\mathcal{M}(D) = \mathcal{R}]} \\
&= \frac{\prod_{i=1}^{n} \Pr[D_i = D'_i \cup \{t_i\}] \cdot \prod_{i=1}^{n} \Pr[M_i(D'_i \cup \{t_i\}) = r_i]}{\prod_{i=1}^{n} \Pr[M_i(D_i) = r_i]}.
\end{aligned} \tag{15}$$

Each mechanism M_i satisfying ρ_i-differential identifiability gives

$$\begin{aligned}
\Gamma(t) &= \frac{\prod_{i=1}^{n} \Pr[D_i = D'_i \cup \{t_i\}] \cdot \prod_{i=1}^{n} \Pr[M_i(D'_i \cup \{t_i\}) = r_i]}{\prod_{i=1}^{n} \Pr[M_i(D_i) = r_i]} \\
&= \prod\nolimits_{i=1}^{n} \frac{\Pr[D_i = D'_i \cup \{t_i\}] \cdot \Pr[M_i(D'_i \cup \{t_i\}) = r_i]}{\Pr[M_i(D_i) = r_i]} \\
&\leqslant \rho_1 \cdot \rho_2 \cdots \rho_n \\
&\leqslant \min \rho_i.
\end{aligned} \tag{16}$$

The sequence of differential identifiability mechanisms $M_i(D_i)$ under parallel composition gives the definition of $(\min \rho_i)$-differential identifiability.

Parallel composition is very important for big data analysis that requires some parallel and aggregation queries in a distributed system. Also, the privacy level of parallel composition that can be fixed is independent of the total number of queries.

6 DI *k*-means clustering algorithms

Lee and Clifton [10] only gave an example that f is a simple aggregate query function about practical applicability of differential identifiability. However, we expect differential identifiability can be applied to more complex privacy preservation issues to evaluate performance, such as data mining in big data. Table 2 [13, 33–37] compares some privacy-preserving k-means algorithms. Currently, differential identifiability can provide an easy parameterization approach. In this section, we combine differential identifiability with k-means algorithm to protect clustering results. Meanwhile, DI k-means algorithm is given on MapReduce framework based on the composition properties of differential identifiability. As a result, the parallel framework can improve the complexity of big data analysis.

6.1 DI *k*-means algorithm on MapReduce

The key to privacy leak is the cluster center in classical k-means algorithms. So it is necessary to only release the approximate values of cluster centers, which can prevent privacy leak and guarantee a certain level of accuracy on clustering. DI k-means algorithm is also based on such idea.

Table 2 Analysis of DP and DI k-means algorithms[a)]

	Algorithm	Characteristic		Complexity	Parameterization
		Initial center points	Clustering		
DP k-means	PADC [33]	Compute the density of data for identifying outliers and divide data into segments for finding k centers	Use the relative distance and add weight to each cluster	$O(ndkT)$	No
	DP k-means based on contour coefficients [34]	Equally divide dataset into k subsets	Clustering on MapReduce, evaluate clusters by contour coefficients and add different noise to k clusters	$O(ndkT)$	No
	Privacy-preserving hybrid k-means [35]	Adopt swarm intelligence to find the optimum centers in distributed data sets	Privacy-preserving k-means	$O(nT_1 + dkT_2)$	No
	DPLk-means [36]	Find initial centers by performing DP k-mean to each subset	DP k-means clustering	$O(ndk'T_1 + ndkT_2)$	No
	Improved DPLloyd [37]	Choose initial centers that is similar to the concept of sphere packing	Optimize the number of rounds and privacy budget allocation	$O(ndkT)$	No
	Optimized canopy DP k-means [13]	Initial centers and the optimal cluster number based on the minimum and maximum principle on MapReduce	DP k-means clustering	$O(Td \max(\max \lvert U_p \rvert, k \max(n_i)))$	No
DI k-means	Not proposed				Yes

a) n is the number of data records, k and k' are the number of clusters, T is the total iteration number, T_1 is the iteration number in the initial stage, T_2 is the iteration number in the clustering stage, $|U_p|$ is the number of data in the pth cluster, and n_i is the number of the ith data chunk

Based on the composition properties of differential identifiability in Section 5, Algorithm 1 shows that DI k-means algorithm can be given on MapReduce framework for big data analysis. Most of the existing k-means algorithms randomly or empirically select initial centers or cluster number. To improve the utility of results, the initial centers and the optimal cluster number are generated by the optimized Canopy algorithm [13]. Mapper class and Reducer class are described in Algorithm 2.

Algorithm 1 MapReduce: DI k-means

Input: A normalized database D with n data records and d dimensions, the number of iterations T, threshold δ, possible worlds number m and privacy parameter ρ.
Output: Cluster centers u'.
Task driver:
  Optimized Canopy generates initial centers $u_1, u_2, \ldots, u_k$ and optimal cluster number k;
  for $(i = 0; i < T; i++)$
    Call Mapper class and generate (s, x_j) key-value pairs;
    Call Reducer class and return centers u' of the current round;
    Calculate the distance centerdist between the current and last round;
    if(centerdist $< \delta$)
      Break;
  Output final cluster centers u'.

Because k-means algorithm is usually used for numerical data, random noise Y is generated by Laplace mechanism. Differential identifiability assumes that an adversary knows each individual information except one. In other words, the number of input database D is fixed. The algorithm can output the accurate number of database D without affecting differential identifiability at all, which is different from DP k-means. By means of Algorithm 1, the database D can release the approximate value of cluster centers to achieve privacy preservation.

When setting multiple Reduce Tasks, each Reduce Task performs independently in each iteration, thus the privacy parameter ρ_i of each iteration is equivalent to parallel combination of Reduce tasks. In other words, if each iteration of the algorithm satisfies ρ_i-differential identifiability, each Reduce task satisfies

Algorithm 2 Mapper and Reducer class: DI k-means

Input: A normalized database D with n data records and d dimensions, initial centers $u_1, u_2, \ldots, u_k$.
Output: Cluster centers u' of current round.

```
Map function:
  Read data record x_j one by one;
  s = 0;
  for (l = 0; l < k; l++)
    Eq. (3) calculates the Euclidean metric dist_l;
    if(dist_l < dist_0)
      s = l;
      dist_0 = dist_l;
  Return s;
  Assign x_j to the nearest center u_s;
  Generate (s, x_j) key-value pairs;
Reduce function:
  Receive (s, x_j) pairs according to s;
  Generate noise Y ~ Lap(λ), λ = Δf / ln((m-1)ρ_i / (1-ρ_i));
  for (p = 0; p < |U_s|; p++)
    num_s = Count((s, x_p));
    sum_s = Sum((s, x_p));
  sum'_s = sum_s + Y;
  u'_s = sum'_s / num_s;
  Return centers u' of current round.
```

ρ_i-differential identifiability.

6.2 Analysis

(1) Performance analysis. Here we give the performance analysis of differential identifiability k-means algorithm on MapReduce. According to classical k-means algorithm, its time complexity is $O(nkTd)$, where n is the number of data records, k is the cluster number, T is the total iteration number, and d is the dimension. The performance of differential identifiability k-means algorithm will greatly reduce for big data analysis. The algorithm on MapReduce framework can improve the complexity of data analysis.

MapReduce divides big data into separate data chunks and processes them in parallel. Assume that MapReduce divides the input database D into M chunks $D_1, \ldots, D_M$, the number of each data chunk is n_i $(1 \leqslant i \leqslant M)$. For Map function in DI k-means algorithm, M data chunks are processed in parallel. The time complexity of Map function is $O(k \max(n_i))$. For Reduce function, MapReduce framework receives the key-value pairs (s, x_j), sorts (s, x_j) according to the value of s, and generates k Reduce Tasks receiving respectively k clusters U_p $(1 \leqslant p \leqslant k)$. Reduce function respectively computes k cluster centers. The time complexity of Reduce function is $O(\max |U_p|)$. Assuming the number of iterations T, the total time complexity of the algorithm is $O(Td \cdot \max(k \max(n_i), \max |U_p|)) < O(nkTd)$. Thus the complexity of the algorithm can be improved by MapReduce framework.

(2) Security analysis. The approximate value of cluster centers is important to achieve security. Random noise Y is the key to generating the approximate value. Δf, m and ρ are privacy parameters for Laplace mechanism to generate noise. For a query function f_{sum}, each dimension of the input database D is normalized to $[0, 1]^d$. The total sensitivity range Δf of the query sequence is at most d.

For the input database D with multiple dimensions, each dimension may have many possible worlds which makes the value of m too large, i.e., the prior probability of an adversary is small. According to sequential composition, with the increasing iterations, ρ_i will gradually approach the prior probability $1/m$ to satisfy $m^{T-1} \prod_{i=1}^{T} \rho_i < 1$. In practice, an adversary may conclude with high confidence whether an individual t is in the database D. Therefore, for numerical databases, assume that m used in the algorithm is $1/\max_{|\Psi|}(\Pr[D = D' \cup \{t\}])$, i.e., the minimum of possible worlds $|\Psi|$ in each attribute. In this case, it is equivalent to assume that the adversary is uncertain about an attribute value of the individual t.

DI k-means algorithm cannot directly set ρ_i of each iteration, which is different from differential privacy. According to sequential composition, the whole algorithm provides $(m^{T-1} \prod_{i=1}^{T} \rho_i)$-differential identifiability. So ρ_i of each iteration needs to be calculated by m, ρ and $m^{\text{iter}-1} \prod_{i=1}^{\text{iter}} \rho_i$ no matter whether the total number of iterations T is fixed or not. Here iter denotes the iterth iteration. Then we need to determine the value of ρ_1.

The experience of k-means algorithm shows that the effect of previous iterations on clustering results is greater than that of later iterations. For differential privacy, Dwork [11] proposed the privacy parameter

Table 3 Description of magic data set

Data set characteristic	Multivariate
Number of instances	19020
Area	Physical
Attribute characteristics	Real
Number of attributes	11
Classification	g = gamma(signal): 12332, h = hadron(background): 6688

strategy that each round iteration consumes half of the remaining privacy budget. The privacy budget for the ith iteration is $\varepsilon_i = \varepsilon/2^i$. Under the equal likeliness between possible worlds assumption, there exists a mapping relationship between two definitions. When ρ and m are fixed, the algorithm also satisfies $\ln\frac{(m-1)\rho}{1-\rho}$-differential privacy. Therefore, ρ_1 can be calculated by $\ln\frac{(m-1)\rho}{1-\rho}/2$. ρ_i of later iterations, which still need hold $\rho_i > 1/m$, will reduce to guarantee $m^{\text{iter}-1}\prod_{i=1}^{\text{iter}}\rho_i \leqslant \rho$. Therefore, for later iterations, $\rho_i = \min(\ln\frac{(m-1)\rho}{1-\rho}/2^i, \frac{\rho}{m^{i-1}\rho_1\cdots\rho_{i-1}})$ can achieve ρ-differential identifiability.

(3) Utility analysis. F-measure [1] is a common index that evaluates the availability of clustering results. F-measure can be calculated by the precision (P) and recall (R) commonly used in data mining results. F-measure represents the degree of the similarity between the clustering results output by clustering algorithms and the standard clustering results. The maximum value of F-measure is 1. Its calculation method is

$$P_i = \frac{\text{cover}_i}{|D_i|}, \quad R_i = \frac{\text{cover}_i}{|C_i|}, \quad F_i = \frac{2P_iR_i}{P_i + R_i}, \quad F = \sum_{i=1}^{k}\frac{|C_i|}{n}F_i, \tag{17}$$

where P_i and R_i ($1 \leqslant i \leqslant k$) are the precision and recall in the ith cluster. C_i and D_i are the ith cluster of data sets C and D. $|C_i|$ and $|D_i|$ are the total number of records in C_i and D_i. cover_i is the total number of the same data records in C_i and D_i. F_i is the F-measure of the ith cluster. n is the number of data records in the data set.

The experimental setting is the Ubuntu 32-bit 16.04LTS system with 2.60 GHz CPU and 8 GB RAM. Eclispe is used as integrated development environment, Hadoop 2.7.3 is used for big data platform, and DI k-means algorithm is implemented by Java programming language. The input database is the "Magic Gamma Telescope Data Set" in the UCI machine learning repository data set. The data set is generated by a Monte Carlo program to simulate registration of high energy gamma particles in a ground-based atmospheric Cherenkov gamma telescope using the imaging technique. The details of magic data set are listed in Table 3.

The magic data set is respectively executed by DP k-means and DI k-means algorithms based on optimized Canopy. The executive program of DI k-means algorithm takes 13.57 s and consumes 244.89 MB of memory on average, while that of DP k-means algorithm takes 16.72 s and consumes 217.96 MB of memory on average. The algorithms take the average of multiple times experiments as the final result of F-measure based on the randomness of added noise. The results output by DP k-means and DI k-means algorithms are shown in Figures 2 and 3.

For the magic data set, $m = 10001$ can be fixed according to the accuracy of normalized data and security analysis. m is also different for the different accuracy of normalized data sets. According to security analysis, DI k-means algorithm can satisfy ρ-differential identifiability. The algorithm can balance the utility and security by selecting ρ.

Figures 2 and 3 show that ε is approximately equal to $\ln\frac{(m-1)\rho}{1-\rho}$ when the clustering results output by DP k-means and DI k-means algorithms are similar. For DI k-means algorithm, it also shows that the privacy parameters of two definitions have a one-to-one mapping relationship under the equal likeliness between possible worlds.

7 DI *k*-prototypes clustering algorithm

DI k-means algorithm is only applicable to numerical data. In fact, there is much mixed data that requires privacy preservation. Privacy-preserving mixed data clustering algorithms are relatively few, especially privacy-preserving k-prototypes. In this section, we combine differential identifiability with k-prototypes algorithm to protect clustering results. Meanwhile, the algorithm is given on MapReduce framework based on exponential mechanism and composition properties of differential identifiability.

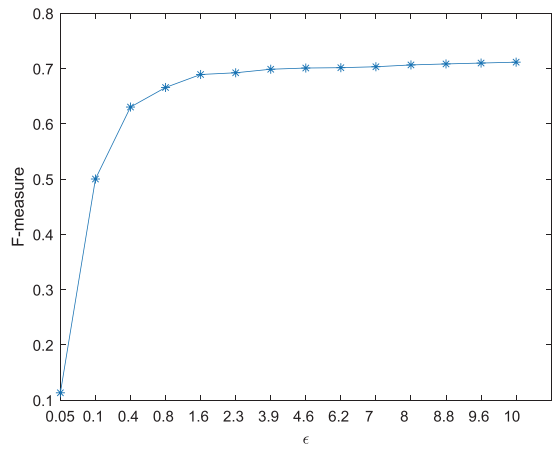

Figure 2 (Color online) F-measure: DP k-means algorithm.

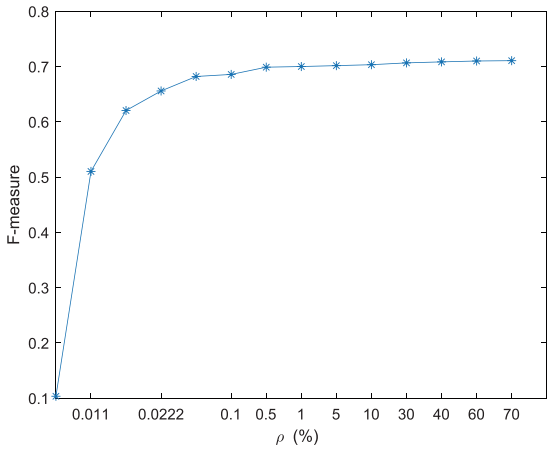

Figure 3 (Color online) F-measure: DI k-means algorithm.

7.1 DI k-prototypes algorithm on MapReduce

The idea of k-prototypes algorithm is similar to that of k-means algorithm. So the basic idea of DI k-prototypes algorithm is also to release the approximate value of cluster centers. As shown in Algorithm 3, DI k-prototypes algorithm is given on MapReduce framework based on the composition properties.

Algorithms 3–5 show that DI k-prototypes algorithm can be used for mixed data. For numerical attributes, random noise Y is still generated by Laplace mechanism. The exponential mechanism of differential identifiability chooses the cluster centers of categorical attributes according to the count of each categorical value, as described in Algorithm 5.

Algorithm 3 MapReduce: DI k-prototypes

Input: A normalized database D with n data records and d dimensions, numerical attributes d_r, categorical attributes d_c, the number of iterations T, threshold δ, possible worlds number m_r, m_c, and privacy parameter ρ^r, ρ^c.
Output: Cluster centers u'.

```
Task driver:
  Optimized Canopy generates initial centers u_1, u_2, ..., u_k and optimal cluster number k;
  for (i = 0; i < T; i++)
    Call Mapper class and generate (s, x_j) key-value pairs;
    Call Reducer class and return centers u' of the current round;
    Calculate the distance centerdist between the current and last rounds;
    if(centerdist < δ)
      Break;
  Output final cluster centers u'.
```

Algorithm 4 Mapper class: DI k-prototypes

Input: A normalized database D with n data records and d dimensions, initial centers $u_1, u_2, \ldots, u_k$.
Output: (s, x_j) key-value pairs.

```
Map function:
  Read data record x_j one by one;
  s = 0; dist_0 = MAX_Value;
  for (l = 0; l < k; l++)
    Eq. (6) calculates the distance dist_l between centers and x_j;
    if(dist_l < dist_0)
      s = l;
      dist_0 = dist_l;
  Return s;
  Assign x_j to the nearest center u_s;
  Generate (s, x_j) key-value pairs.
```

7.2 Analysis

(1) Security analysis. Performance analysis is similar to DI k-means algorithm and is not described here. DI k-prototypes algorithm uses Laplace mechanism and exponential mechanism to respectively generate

Algorithm 5 Reducer class: DI k-prototypes

Input: (s, x_j) key-value pairs.
Output: Cluster centers u' of current round.
Reduce function:
Receive (s, x_j) pairs according to s;
for $(l = 0;\ l < d_r;\ l++)$
for $(p = 0;\ p < |U_s|;\ p++)$
$\text{num}_s = \text{Count}((s, x_p))$,
$\text{sum}_s^r = \text{Sum}((s, x_p))$;
Generate noise $Y \sim \text{Lap}(\lambda)$, $\lambda = \frac{\Delta f}{\ln \frac{(m_r-1)\rho_i^r}{1-\rho_i^r}}$;
$\text{sum}_{s'}^r = \text{sum}_s^r + Y$;
$u_{s'}^r = \text{sum}_{s'}^r / \text{num}_s$;
for $(l = 0;\ l < d_c;\ l++)$
for $(p = 0;\ p < |U_s|;\ p++)$
Score function sf counts the number of each categorical value;
DI exponential mechanism chooses the categorical value as center $u_{s'}^c$ with the probability proportional to $\exp(\text{sf} \ln \frac{(m_c-1)\rho_i^c}{1-\rho_i^c}/2\Delta\text{sf})$;
Return centers u' of current round.

numerical and categorical centers. Based on parallel composition of differential identifiability, the input database D is considered to consist of two disjoint subsets, i.e., numerical subset and categorical subset. Privacy parameters Δf, m and ρ are selected respectively. The privacy preservation level ρ of the input database D can be fixed by parallel composition. For the numerical subset, each dimension of the input database D is normalized to $[0, 1]^d$. Δf^r of the query function f_{sum} is at most d_r. Δsf of the score function sf is 1 for the categorical subset.

For the input mixed database D with multiple dimensions, the selection of m is similar to that of DI k-means algorithm. m_r and m_c are the minimum of possible worlds $|\Psi_r|$ and $|\Psi_c|$, respectively. $\rho^r > 1/m_r$ is the privacy level of numerical subset and $\rho^c > 1/m_c$ is the privacy level of categorical subset. In each iteration, ρ_i^r is $\min(\ln \frac{(m_r-1)\rho^r}{1-\rho^r}/2^i, \frac{\rho^r}{m_r^{i-1}\rho_1^r \cdots \rho_{i-1}^r})$ and ρ_i^c is $\min(\ln \frac{(m_c-1)\rho^c}{1-\rho^c}/2^i, \frac{\rho^c}{m_c^{i-1}\rho_1^c \cdots \rho_{i-1}^c})$ to achieve ρ^r-differential identifiability and ρ^c-differential identifiability. For the input database D, DI k-prototypes algorithm satisfies $\min(\rho^r, \rho^c)$-differential identifiability based on parallel composition.

(2) Utility analysis. DI k-prototypes algorithm also uses the index F-measure to evaluate the availability of clustering results. The experimental setting is the same as DI k-means algorithm. The input database is the "Statlog(Heart) Data Set" in the UCI machine learning repository data set. The data set is a heart disease database similar to already present in the repository. Heart data set has 270 instances, each of which has 14 attributes. All instances can be divided into two clusters: absence or presence of heart disease. 13 of 14 attributes represent pathological examination. The attribute description of heart data set is listed in Table 4.

Eq. (6) in k-prototypes algorithm has a weight γ, which is crucial for whole clustering process. The weight γ causes the measure of similarity to be biased towards a certain attribute type. Therefore, the best value of γ should be fixed experimentally. The heart data set is implemented by k-prototypes algorithm based on optimized Canopy. The experimental results of Figure 4 show that F-measure has the maximum value when $\gamma = 0.12$.

The heart data set is respectively executed by DP k-prototypes and DI k-prototypes algorithms based on optimized Canopy. The executive program of DI k-prototypes algorithm takes 8.07 s and consumes 394.07 MB of memory on average, while that of DP k-prototypes algorithm takes 8.02 s and consumes 385.02 MB of memory on average. The algorithms also take the average of multiple times experiments as the final result of F-measure. The results output by DP k-prototypes and DI k-prototypes algorithms are shown in Figures 5 and 6, respectively.

For the normalized numerical subset of the heart data set, $m_r = 10001$ can be fixed according to the accuracy of data and security analysis. $m_c = 2$ is also fixed according to the minimum of categorical attributes.

According to security analysis, the algorithm can satisfy $\rho^r > 10^{-4}$-differential identifiability and $\ln \frac{(m_r-1)\rho^r}{1-\rho^r}$-differential privacy for numerical subset. For the categorical subset, the algorithm can satisfy $\rho^c > 0.5$-differential identifiability and $\ln \frac{(m_c-1)\rho^c}{1-\rho^c}$-differential privacy. Based on composition properties, the heart data set satisfies $\min(\rho^r, \rho^c)$-differential identifiability and $\max(\ln \frac{(m_r-1)\rho^r}{1-\rho^r}, \ln \frac{(m_c-1)\rho^c}{1-\rho^c})$-differential privacy.

Table 4 Attribute description of heart data set

Attribute	Characteristic	Information
Age	Numerical	–
Sex	Categorical	1: male; 2: female
Chest pain type	Categorical	1: typical angina; 2: atypical angina; 3: non-angina pectoris; 4: asymptomatic
Resting blood pressure	Numerical	–
Serum	Numerical	mg/dl
Fasting blood Sugar	Categorical	0: abnormal($>$ 120 mg/dl); 1: normal
Resting electrocardiographic results	Categorical	0: normal; 1: ST-T abnormal; 2: left ventricular hypertrophy
Maximum heart rate achieved	Numerical	–
Exercise induced angina	Categorical	0: no; 1: yes
ST depression induced by exercise relative to rest	Numerical	–
The slope of the peak exercise ST segment	Categorical	1: rise; 2: steady; 3: decline
Number of major vessels	Numerical	
Thalassemia	Categorical	3: normal; 6: fixed defect; 7: reversable defect

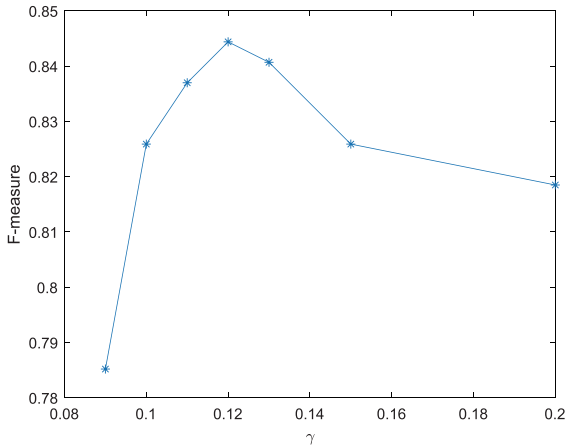

Figure 4 (Color online) F-measure: the weight γ.

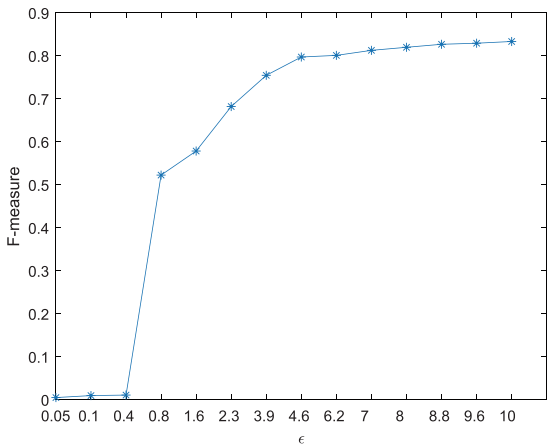

Figure 5 (Color online) F-measure: DP k-prototypes algorithm.

Figure 6 shows that the algorithm can balance the utility and security when $\rho^r = 0.05$ and $\rho^c = 0.7$. According to the mapping relationship between privacy parameters, the algorithm satisfies ($\varepsilon = 6.26$)-differential privacy. From Figure 5, it can be seen that the two clustering results are close when $\varepsilon = 6.2$.

8 Conclusion

In this paper, we proposed the exponential mechanism of differential identifiability that is used for non-numerical data, and proved the composition properties which can allow us to apply differential identifiability for complex privacy preservation queries in big data analysis. Based on the composition and exponential mechanism, we combined differential identifiability with k-means and k-prototypes al-

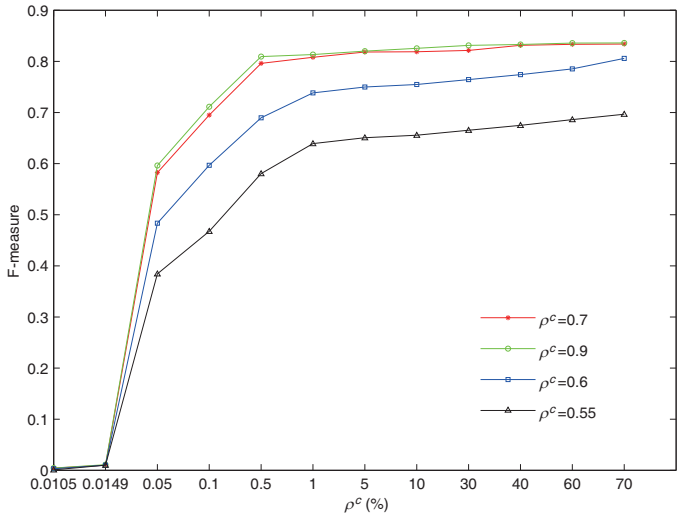

Figure 6 (Color online) F-measure: DI k-prototypes algorithm.

gorithms. The algorithms were implemented on MapReduce for big data analysis. Performance analysis shows that MapReduce framework divides input database into several data chunks, which can improve the complexity of big data analysis. And security analysis shows that the algorithms satisfy ρ-differential identifiability. By comparing the clustering results of two definitions in utility analysis, DI k-means and DI k-prototypes algorithms satisfy the mapping relationship between privacy parameters, which can provide a maximum value of ε based on the number of possible worlds m and the given ρ. Our achievements provide basis for practical application of differential identifiability in big data analysis, and provide an easy parameterization approach for big data analysis practitioners. Furthermore, we will verify the algorithms by means of more databases and expect that more big data analysis algorithms with differential identifiability may support numerical and non-numerical data.

Acknowledgements This work was supported by National Key Research and Development Program of China (Grant No. 2016YFC1000307) and National Natural Science Foundation of China (Grant Nos. 61971021, 61571024).

References

1 Mendes R, Vilela J P. Privacy-preserving data mining: methods, metrics, and applications. IEEE Access, 2017, 5: 10562–10582
2 Yu S. Big privacy: challenges and opportunities of privacy study in the age of big data. IEEE Access, 2016, 4: 2751–2763
3 Dinur I, Nissim K. Revealing information while preserving privacy. In: Proceedings of ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, San Diego, 2003. 202–210
4 Dwork C, Nissim K. Privacy-preserving datamining on vertically partitioned databases. In: Advances in Cryptology – CRYPTO 2004. Berlin: Springer, 2004. 528–544
5 Blum A, Dwork C, McSherry F, et al. Practical privacy: the SuLQ framework. In: Proceedings of ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, Baltimore, 2005. 128–138
6 Dwork C, McSherry F, Nissim K, et al. Calibrating noise to sensitivity in private data analysis. In: Theory of Cryptography. Berlin: Springer, 2006. 265–284
7 Dwork C. Differential privacy. In: Automata, Languages, and Programming. Berlin: Springer, 2006. 1–12
8 Kifer D, Machanavajjhala A. No free lunch in data privacy. In: Proceedings of ACM SIGMOD International Conference on Management of Data, Athens, 2011. 193–204
9 Cormode J. Personal privacy vs population privacy: learning to attack anonymization. In: Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, 2011. 1253–1261
10 Lee J, Clifton C. Differential identifiability. In: Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Beijing, 2012. 1041–1049
11 Dwork C. A firm foundation for private data analysis. Commun ACM, 2011, 54: 86–95
12 Li L, Dong Y, Wang J. Differential privacy data protection method based on clustering. In: Proceedings of Cyber-Enabled Distributed Computing and Knowledge Discovery, Nanjing, 2017. 11–16
13 Zhao Z, Shang T, Liu J W, et al. Clustering algorithm for privacy preservation on MapReduce. In: Proceedings of the 4th International Conference on Cloud Computing and Security, Haikou, 2018. 622–632
14 Bugata P, Drotar P. On some aspects of minimum redundancy maximum relevance feature selection. Sci China Inf Sci, 2020, 63: 112103
15 Samarati P, Sweeney L. Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression. In: Proceedings of IEEE Symposium on Security and Privacy, Oakland, 1998. 1–19
16 Samarati P. Protecting respondents identities in microdata release. IEEE Trans Knowl Data Eng, 2001, 13: 1010–1027
17 Sweeney L. k-anonymity: a model for protecting privacy. Int J Unc Fuzz Knowl Based Syst, 2002, 10: 557–570
18 Machanavajjhala A, Gehrke J, Kifer D, et al. l-diversity: privacy beyond k-anonymity. In: Proceedings of International Conference on Data Engineering, Atlanta, 2006. 24–35

19 Li N, Li T, Venkatasubramanian S. t-closeness: privacy beyond k-anonymity and l-diversity. In: Proceedings of International Conference on Data Engineering, Istanbul, 2007. 106–115
20 Li N H, Li T C, Venkatasubramanian S. Closeness: a new privacy measure for data publishing. IEEE Trans Knowl Data Eng, 2010, 22: 943–956
21 Li N H, Qardaji W, Su D, et al. Membership privacy: a unifying framework for privacy definitions. In: Proceedings of ACM SIGSAC Conference on Computer and Communications Security, New York, 2013. 889–900
22 Backes M, Berrang P, Humbert M, et al. Membership privacy in microRNA-based studies. In: Proceedings of ACM SIGSAC Conference on Computer and Communication, Vienna, 2016. 319–330
23 Aggarwal G, Feder T, Kenthapadi K, et al. Achieving anonymity via clustering. In: Proceedings of ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, Chicago, 2006. 153–162
24 Byun J W, Kamra A, Bertino E, et al. Efficient k-anonymization using clustering techniques. In: Proceedings of International Conference on Database Systems for Advanced Applications, Bangkok, 2007. 188–200
25 Nissim K, Raskhodnikova S, Smith A. Smooth sensitivity and sampling in private data analysis. In: Proceedings of ACM Symposium on Theory of Computing, San Diego, 2007. 75–84
26 Jafer Y, Matwin S, Sokolova M. Using feature selection to improve the utility of differentially private data publishing. Procedia Comput Sci, 2014, 37: 511–516
27 Huang Z. Clustering large data sets with mixed numeric and categorical values. In: Proceedings of Pacific-Asia Conference on Knowledge Discovery and Data Mining, Singapore, 1997. 21–34
28 Huang Y. Understanding of Big Data: Big Data Processing and Programming Practices. Beijing: Machinery Industry Press, 2014
29 McSherry F, Talwar K. Mechanism design via differential privacy. In: Proceedings of IEEE Annual Symposium on Foundations of Computer Science, Providence, 2007. 94–103
30 Dwork C, Rothblum G N, Vadhan S. Boosting and differential privacy. In: Proceedings of IEEE Annual Symposium on Foundations of Computer Science, Las Vegas, 2010. 51–60
31 Bkakria A, Cuppens-Boulahia N, Cuppens F. Linking differential identifiability with differential privacy. In: Proceedings of International Conference on Information and Communications Security, Cham, 2018. 232–247
32 McSherry F. Privacy integrated queries:an extensible platform for privacy-preserving data analysis. Commun ACM, 2010, 53: 89–97
33 Xiong J, Ren J, Chen L, et al. Enhancing privacy and availability for data clustering in intelligent electrical service of IoT. IEEE Internet Things J, 2019, 6: 1530–1540
34 Zhang Y, Liu N, Wang S. A differential privacy protecting k-means clustering algorithm based on contour coefficients. Plos One, 2018, 13: e0206832
35 Gao Z, Sun Y, Cui X, et al. Privacy-preserving hybrid k-means. Int J Data Warehous Min, 2018, 14: 1–17
36 Ren J, Xiong J, Cui X, et al. DPLk-means: a novel differential privacy k-means mechanism. In: Proceedings of IEEE International Conference on Data Science in Cyberspace, Shenzhen, 2017. 133–139
37 Su D, Cao J, Li N, et al. Differentially private k-means clustering and a hybrid approach to private optimization. ACM Trans Priv Secur, 2017, 20: 1–33