Density Peak Clustering Algorithm Based on Differential Privacy Preserving

Yun Chen, Yunlan Du and Xiaomei Cao

September 29, 2019

Density Peak Clustering Algorithm Based on Differential Privacy Preserving

Yun Chen[1], Yunlan Du[2], Xiaomei Cao[1]

[1] School of Computer Science, Nanjing University of Posts and Telecommunications,
Nanjing, 210046, China
[2] Department of Computer Science and Technology, Nanjing University,
Nanjing, 210046, China
caoxm@njupt.edu.cn

Abstract. Clustering by fast search and find of density peaks (CFSFDP) is an efficient algorithm for density-based clustering. However, such algorithm inevitably results in privacy leakage. In this paper, we propose DP-CFSFDP to address this problem with differential privacy, which adds random noise in order to distort the data but preserve its statistical properties. Besides, due to the poor performance of CFSFDP on evenly distributed data, we further optimize the clustering process with reachable-centers and propose DP-rcCFSFDP. The experimental results show that, under the same privacy budget, DP-rcCFSFDP can improve the clustering effectiveness while preserving data privacy compared with DP-CFSFDP.

Keywords: Differential Privacy, Clustering, Density Peak, Privacy Preserving.

1 Introduction

In the era of big data, the launches of services and products are relying more on the user data (i.e. privacy) and information mined from it. As data privacy is inevitably exposed in the process of data collection, analysis and publication, privacy protection technology is developed to address these privacy threats. Recently, many privacy protection methods based on k-anonymity [1,2] and partition [3,4] have emerged. Although these methods can protect more details of data, they all under special attack assumptions.

Differential privacy is an innovative conception demonstrated by Dwork [5,6,7] for privacy leakage of statistical databases. With random noise, it distorts the sensitive data and preserves the privacy from the malicious attackers. This technique inspires researchers to introduce appropriate noise to data and arm clustering analyses with differential privacy correspondingly.

For example, Blum et al. [8] first introduced differential privacy into clustering analysis. They improved a k-means clustering algorithm and perturbed the query response to protect each database entry. Wu et al. [9] then applied differential privacy technique to density-based clustering algorithm for the first time and proposed DP-

DBSCAN algorithm. Though the clustering methods with differential privacy are improving year by year, the algorithms are still limited by the unsuitability of clusters with complex shapes [8,10,11] and the sensitiveness to input parameters [9,12,13,14].

In this paper, we leverage a more efficient density peak clustering algorithm, clustering by fast search and find of density peaks (CFSFDP) [20], which clusters data by connecting points to the nearest and denser points, and propose an improved DP-CFSFDP by introducing differential privacy protection to it, aiming at solving privacy leakage problem. We add Laplacian noise depending on the differential privacy mechanism when the Gaussian kernel function is called during the density calculation. Due to the poor performance of CFSFDP on data with uniform distribution, an improved algorithm with reachable-centers (DP-rcCFSFDP) is proposed to optimize the clustering process. We allow the lower-density center points to cluster with the reachable and higher-density center points, thus DP-rcCFSFDP can improve the effectiveness of clustering while satisfying the requirement of security.

2 Background and Related Work

2.1 Differential Privacy

Differential privacy preserving is a technique to protect private data by adding random noise to sensitive data while maintaining the data attributes or their statistical properties [5]. We suppose the attacker has obtained all data except the target data. With differential privacy preserving, he still cannot obtain the target. The definitions of differential privacy are as follows.

Definition 1. Suppose D and D' are any pair of neighboring datasets that differ by at most one piece of data, M is a randomized algorithm, $Pr[X]$ is the disclosure risk of event X, and $S \subseteq Range(M)$ is the output of algorithm M. If the algorithm M satisfies:

$$Pr[M(D) \in S] \leq e^{\varepsilon} \times Pr[M(D') \in S] \tag{1}$$

Then the algorithm M is said to be ε-differentially private [5]. ε denotes the privacy protection parameter, also known as the privacy budget. The smaller the ε is, the more noise is added, and the more privacy protection is provided.

Definition 2. For the query function $f : D \rightarrow D^d$, its sensitivity [6] Δf is defined as:

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

where $\|\cdot\|_1$ denotes the first-order norm distance.

Differential privacy works by adding noise perturbations. There are two common noise addition mechanisms: Laplace mechanism [7] for numerical data and Exponential mechanism [15] for non-numeric data. The amount of noise depends on sensitivity and privacy budget. In this paper, we implement differential privacy with Laplace mechanism.

Definition 3. Given a dataset *D*, a function *f* with sensitivity Δ*f*, and privacy budget *ε*, thus the randomized algorithm *M* (*D*):

$$M(D) = f(D) + Lap\left(\frac{\Delta f}{\varepsilon}\right) \tag{3}$$

provides *ε*-differential privacy preserving [7]. The *Lap* (Δ*f* / *ε*) is a random noise of Laplace distribution.

Let *b* denote the scale parameter Δ*f* / *ε*, the probability density function of the Laplace distribution is:

$$p(x) = \frac{1}{2b} exp\left(-\frac{|x|}{b}\right) \tag{4}$$

2.2 CFSFDP Algorithm

The main idea of CFSFDP is that each class has a maximum density point as the center point which attracts and connects the lower density points around it, while different class centers are far away from each other. The algorithm defines two quantities: local density ρ_i and distance δ_i.

Definition 4. ρ_i denotes the local density, and there are two calculation methods: based on the cutoff kernel and based on the Gaussian kernel. The local density of x_i calculated by cutoff kernel is defined as:

$$\rho_i = \sum_j \chi(d_{ij} - d_c) \tag{5}$$

where d_{ij} denotes the Euclidean distance between x_i and x_j, d_c denotes the cutoff distance, and ρ_i denotes the number of all remaining points contained in the circle with point x_i as the center and d_c as the radius.

When the data distribution of the dataset is uniform, Eq. (5) may make different points with the same local density, which affects the subsequent cluster calculation. For this reason, another method is proposed for calculating the local density ρ_i with Gaussian kernel function:

$$\rho_i = \sum_j e^{-\left(\frac{d_{ij}}{d_c}\right)^2} \tag{6}$$

In this paper, Gauss kernel function is used to calculate local density.

Definition 5. Distance δ_i denotes the minimum distance between point x_i and other points with higher density, and the equation is as follows:

$$\delta_i = \begin{cases} \min\limits_{j:\rho_j > \rho_i} \left\{d_{ij}\right\} \\ \max\limits_{j} \left\{d_{ij}\right\}, \ otherwise \end{cases} \tag{7}$$

When point x_i has the maximum local density, δ_i denotes the distance between x_i and the point with the maximum distance from x_i.

The CFSFDP selects cluster centers by the decision graph. The decision graph takes ρ as the abscissa and δ as the ordinate. When the point has both larger values of ρ and δ, it is considered as the cluster center. An instructive measurement for choosing the number of centers is provided by the plot of $\gamma_i = \rho_i \cdot \delta_i$ sorted in decreasing order [20]. The remaining points are connected to the nearest point corresponding to their δ_i for clustering.

3 CFSFDP Algorithm Based on Differential Privacy

3.1 DP-CFSFDP

CFSFDP algorithm selects k cluster centers according to the decision graph. The rest points are arranged in descending order of local density and gradually connected to the nearest point with higher density until to a center point. The algorithm performs well on datasets with different shapes or uneven density distribution. However, the density of points may expose the distribution of dataset. The density peak clustering algorithm based on differential privacy preserving (DP-CFSFDP) introduces Laplacian noise to the function of local density calculation, in order to accord with the ε-differentially private and avoid the risk of privacy leakage caused by local density.

The steps of DP-CFSFDP are as follows:

First, initialize the quantities of each point - ρ_i' and δ_i. Calculate the Euclidean distance between points and local density ρ_i. Based on sensitivity and privacy budget, we generate random noise corresponding to Laplace distribution and add it to the density ρ_i. The new densities ρ_i' are arranged in descending order. Thus, we calculate δ_i which indicates the distance from point i to its nearest point with a larger local density.

Second, generate the decision graph based on density ρ_i' and distance δ_i, thereby determine the class centers.

Finally, cluster non-central points. We traverse the rest points in descending order of density, and classify each point and its nearest point with distance δ_i into a class.

The pseudo code of the DP-CFSFDP algorithm is presented in the Algorithm 1.

```
Algorithm 1 DP-CFSFDP

Input: data set $D$, cutoff distance $d_c$, privacy budget $\varepsilon$
Output: clustering results with differential privacy
Calculate $\rho_i$ from Eq.(6) on $D$,
and generate its descending-order subscript $q_i$
$b=\Delta f/\varepsilon$, Generate random noise $Lap(b)$
$\rho_i' = \rho_i + Lap(b)$
Calculate $\delta_i$ from Eq.(7), and generate
its corresponding subscript $n_i$
Draw the decision graph based on $\rho_i'$ and $\delta_i$
Select the appropriate class centers $m_j$,
initialize the clustering label $C_i$=-1
for $i$ = 1:$j$ do
    $C_{mj}$ = $i$
end for
for $i$ = 1:$N$ do
    if point $q_i$ is not classified
        $C_{qi}$ = $C_{nqi}$
    end if
end for
```

3.2 DP-CFSFDP with reachable-centers

DP-CFSFDP algorithm protects data privacy by introducing noise into local density. However, the arrangement order of local density may change due to the added Laplacian noise, and then interfere with the calculation of the distance δ resulting in the change in the distribution of the decision graph. Since the center points is generated from the decision graph, the parameters with noise may lead to the deviation between the new center point and the correct one. Besides, points are likely to be misclassified under the influence of noise during the clustering.

In addition, CFSFDP algorithm supposes that each class must be a maximum density point as the class center. If the density distribution of a class is uniform, or there are multiple distant points with high density, an entire class will be divided into several subclasses. CFSFDP algorithm selects k centers based on the decision graph. However, the inappropriate number of centers may have a great impact on the clustering results.

In this paper, DP-CFSFDP algorithm with reachable-centers (DP-rcCFSFDP) is proposed to reduce the influence of Laplacian noise on clustering results, optimize the selection of centers and make up for the inapplicability of CFSFDP algorithm to uniformly distributed data. The improved algorithm refers to some ideas of DBSCAN [21] and defines *reachable*, and applies it to the classification of the center points. The definitions used in DP-rcCFSFDP are as follows:

Neighbors. The neighbors of x_i are all points in the neighborhood with x_i as the center and *eps* as the radius. In our algorithm, the cutoff distance d_c is used as *eps* to represent the radius of neighborhood.

Reachable. There is a series of points p_1, p_2, p_3... p_m, p_m is said to be reachable from p_i if each p_{i+1} lies in the neighborhood of p_i.

The specific steps of DP-rcCFSFDP are as follows:

First, initialize the quantities ρ_i' and δ_i, and generate the decision graph. This process is the same as the beginning of DP-CFSFDPs.

Second, we select *k_init* points as the initial centers according to the decision graph. We then calculate the delta-density value of gamma by $\gamma_i = \rho_i' \cdot \delta_i$, and arrange them in descending order. The *k_init* points with the largest gamma are selected as initial cluster centers points.

Third, the initial centers are arranged in descending order of density for traversal processing. If the center point with higher density is reachable from a point with lower density with respect to d_c, the lower one will be classified into the cluster of the higher one. We will obtain the accurate number of centers *k* after the traversal.

Finally, the remaining points are traversed in descending order of density, and classified to the cluster of the nearest point with higher density until each of them is connected to a class center. The clustering results will be printed at last.

The cluster process of DP-rcCFSFDP algorithm is presented in the Algorithm 2.

```
Algorithm 2 DP-rcCFSFDP

Input: data set $D$, cutoff distance $d_c$, privacy budget $\varepsilon$
Output: clustering results with differential privacy
Calculate $\rho_i$ from Eq.(6) on $D$,
   and generate its descending-order subscript $q_i$
$b = \Delta f / \varepsilon$, Generate random noise $Lap(b)$
$\rho_i' = \rho_i + Lap(b)$
Calculate $\delta_i$ from Eq.(7), and generate
   its corresponding subscript $n_i$
Draw the decision graph based on $\rho_i'$ and $\delta_i$,
   and calculate $\gamma_i = \rho_i' \cdot \delta_i$ in descending order
Calculate the neighbors of each point based on $d_c$
Select $k\_init$ points with the largest $\gamma$
   as the initial cluster centers
Initialize class count $nc$=1
The initial centers are sorted in descending order
   of density $Clist_m$, and $Clist_1$ is the $nc$ class
for $i$ = 1:$m$ do
    for $j$ = 1:$i$ do
        if $Clist_j$ is reachable from $Clist_i$ w.r.t. $d_c$
           $Clist_i$ is classified to $Clist_j$
           break
        end if
    end for
    if $Clist_i$ is not classified
        $nc$=$nc$+1
        $Clist_i$ is the $nc$ class
    end if
end for
Non-central points are arranged according to $q_i$,
and classified to the class of $n_i$
```

DP-CFSFDP algorithm is sensitive to the selection of centers. Though the number of center points meets the actual clustering requirements, the selection of centers will still be interfered with Laplacian noise, resulting in biased centers or even multiple centers in one class. While DP-rcCFSFDP selects *k_init* points as initial centers (*k_init* is greater than or equal to the number of actual centers number), it classifies the reachable centers into one class, which finally corrects the biased center points generated by noise to connect to the right one. The algorithm reduces the dependence on the number of centers, reduces the interference of noise on clustering, and improves the stability.

3.3 Privacy Analysis

According to Eq. (6) of local density and Definition 2 of sensitivity, the sensitivity of the local density function is 1 when a point is added or deleted in the normalized space $[0, 1]^d$.

Suppose that two datasets $D1$ and $D2$ differ by at most one record, $M\left(D1\right)$ and $M\left(D2\right)$ denote the output of CFSFDP algorithm with Laplacian noise on $D1$ and $D2$, S denotes the arbitrary output, $f\left(D1\right)$ and $f\left(D2\right)$ denote the true clustering results on these datasets, and $s\left(x\right)$ denotes a certain clustering result. According to Eq. (2) and Eq. (4), the security proof of DP-CFSFDP and DP-rcCFSFDP is as follows:

$$
\begin{aligned}
\frac{Pr\left[M\left(D1\right)\in S\right]}{Pr\left[M\left(D2\right)\in S\right]} &= \frac{exp\left(-\frac{\varepsilon\left|\,f\left(D1\right)-s\left(x\right)\right|}{\Delta f}\right)}{exp\left(-\frac{\varepsilon\left|\,f\left(D2\right)-s\left(x\right)\right|}{\Delta f}\right)} \\
&= exp\left(\frac{\varepsilon\left(\left|\,f\left(D2\right)-s\left(x\right)\right|-\left|\,f\left(D1\right)-s\left(x\right)\right|\right)}{\Delta f}\right) \\
&\le exp\left(\frac{\varepsilon\left|\,f\left(D2\right)-f\left(D1\right)\right|}{\Delta f}\right) \\
&= exp\left(\frac{\varepsilon\left\|\,f\left(D2\right)-f\left(D1\right)\right\|_1}{\Delta f}\right) \\
&\le exp\left(\varepsilon\right)
\end{aligned}
$$

The first inequality follows from the triangle inequality which indicates the difference between any two sides is less than the third. According to Definition 1, it is proved that DP-CFSFDP and DP-rcCFSFDP are ε-differentially private.

4 Experiments

4.1 Experiment Setup

The proposed algorithms are implemented in the Python language. The experiments are conducted on a computer with win10 x64 system, Intel i7-6700HQ @2.60GHz CPU and 8GB RAM. The datasets used are from the artificial datasets [22] and UCI Knowledge Discovery Archive database [23].

The specific information of the datasets is shown in Table 1.

Table 1. datasets information

Datasets	Instances	Dimensions	Clusters
Jain	373	2	2
Wine	178	13	2
Aggregation	788	2	7
Iris	150	4	3

4.2 Evaluation Criteria

F-measure [24] and adjusted Rand index (ARI) [25] are used to compare the similarity between the clustering results of proposed algorithms and the ground truth class assignment to evaluate the clustering effectiveness. F-measure is the harmonic average of the precision and recall. ARI is to measure the similarity of the two assignments.

Suppose that T_j is the class in the real clustering results, and D_i the clustering results output from the algorithm proposed in the paper. N is the total number of points in the dataset. $|T_j|$ and $|D_i|$ denote the number of points in the class. The rate of precision, recall and the value of F-measure of T_j and D_i are defined as follows:

$$P\left(T_j, D_i\right) = \frac{\left|T_j \cap D_i\right|}{|D_i|} \tag{8}$$

$$R\left(T_j, D_i\right) = \frac{\left|T_j \cap D_i\right|}{\left|T_j\right|} \tag{9}$$

$$F\left(T_j, D_i\right) = \frac{2 \cdot P\left(T_j, D_i\right) \cdot R\left(T_j, D_i\right)}{P\left(T_j, D_i\right) + R\left(T_j, D_i\right)} \tag{10}$$

F-measure of the clustering results is the weighted average of F-measure for all clusters:

$$F\text{-}measure = \sum_j \frac{\left|T_j\right|}{N} \max_i F\left(T_j, D_i\right) \tag{11}$$

ARI is the improvement of Rand index (RI). Variations of the ARI account for different models of random clustering [26]. Suppose that *T* is the actual clustering results, *D* is the clustering results obtained by the improved algorithm, *a* is the number of pairs of elements that are in the same set in *T* and *D*, and *b* be the number of pairs of elements that are in different sets in *T* and *D*. *E*[*RI*] denotes the expectation of RI, then RI and ARI are defined:

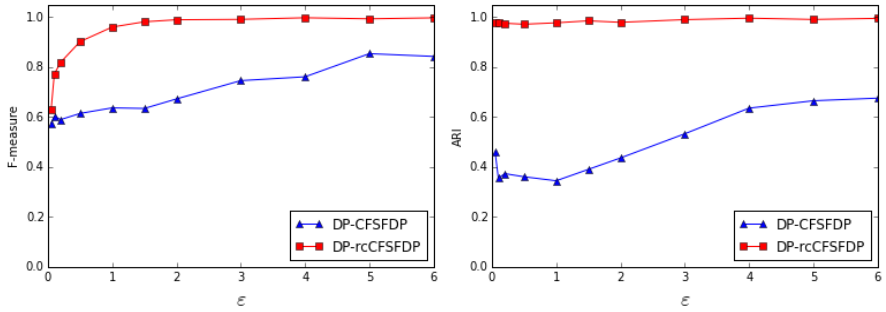
$$RI = \frac{a + b}{C_N^2} \tag{12}$$

$$ARI = \frac{RI - E[RI]}{max(RI) - E[RI]} \tag{13}$$

The range of F-measure is [0,1] and ARI is [-1,1]. The higher the value is, the more similar the outputs of clustering algorithm are to the real clustering results and the less the impact of Laplacian noise on clustering effectiveness.

4.3 Results and Discussion

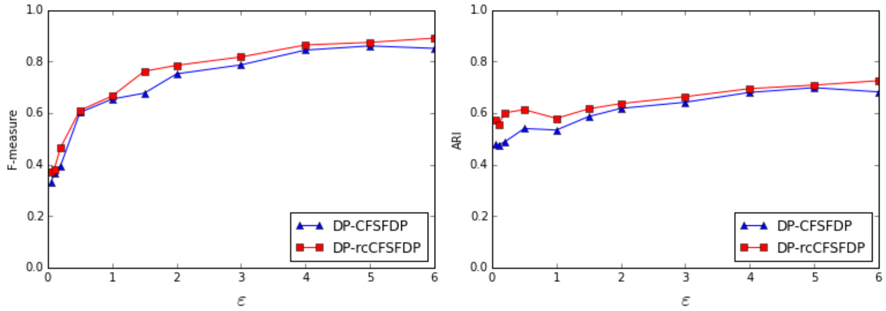
In the experiment, the datasets are normalized so that each attribute value is limited to [0, 1]. To achieve the best clustering effect, appropriate parameters should be selected before we add the noise. DP-CFSFDP and DP-rcCFSFDP are applied on four datasets. For each privacy budget and each metric, we apply the algorithms on each dataset for 30 times and compute their average performances. When the privacy budget *ε* changes, the F-measure and ARI values of the clustering results are shown in the figure.



(a) F-measure on Jain dataset

(b) ARI on Jain dataset

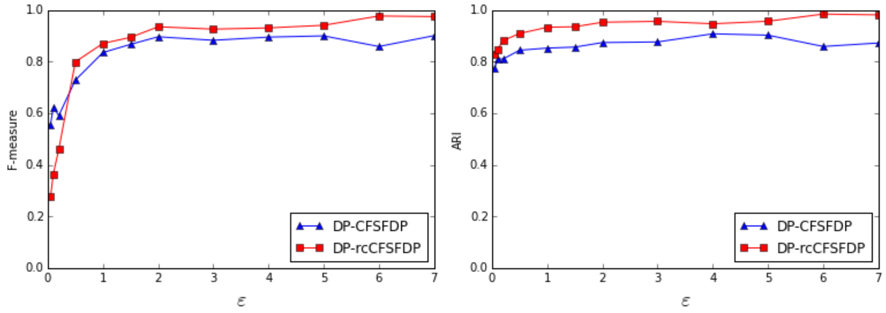
Fig. 1. F-measure and ARI comparison of algorithms on Jain dataset



(a) F-measure on Wine dataset

(b) ARI on Wine dataset

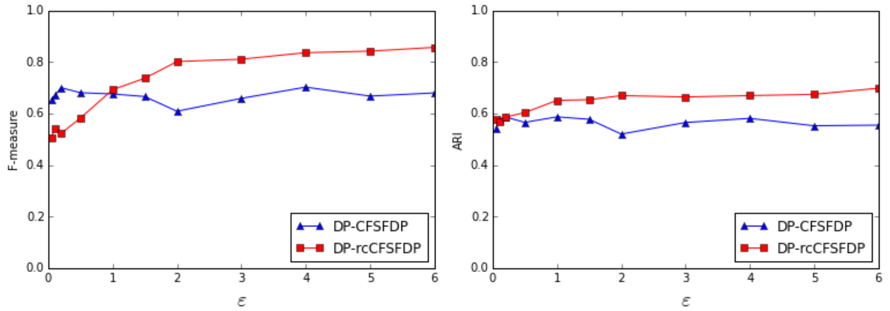
Fig. 2. F-measure and ARI comparison of algorithms on Wine dataset



(a) F-measure on Aggregation dataset

(b) ARI on Aggregation dataset

Fig. 3. F-measure and ARI comparison of algorithms on Aggregation dataset



(a) F-measure on Iris dataset (b) ARI on Iris dataset

Fig. 4. F-measure and ARI comparison of algorithms on Iris dataset

The left side of the figures (Figures. 1(a) - 4(a)) depicts **F-measure** of the clustering results. As is shown, with the growth of privacy budget, F-measure gradually increases and tends to be stable. Since the privacy budget is inversely proportional to the size of the Laplacian noise, the higher the privacy budget, the less the noise and the better the clustering results.

When we compare the performance of DP-CFSFDP and DP-rcCFSFDP under the same privacy budget, it is easy to find that in Fig. 1(a) and Fig. 2(a), DP-rcCFSFDP always has a higher F-measure value than DP-CFSFDP, and that the clustering result is closer to the real result. However, in Fig. 3(a) and Fig. 4(a), when the privacy budget takes a small value, the F-measure value of DP-CFSFDP becomes higher but seems more unstable. The reason is that when the privacy budget is small, too much noise leads to the increasing randomness of the centers selection by DP-CFSFDP algorithm and coincidentally generates even better centers than the original algorithm. When the privacy budget takes a larger value, the clustering of DP-rcCFSFDP is of higher accuracy and more stable, resulting from the optimization of center points classification with the reachable centers.

The right side of the figures (Figures. 1(b) - 4(b)) depicts **ARI** of the clustering results. As we can see, ARI gradually increases and then flattens with the increase of privacy budget. Under the same privacy budget, the ARI value of DP-rcCFSFDP is generally superior than DP-CFSFDP, since the calculation of ARI ignores permutations. Thus, under the same level of privacy protection, the similarity between clustering results of DP-rcCFSFDP and real ones is higher, indicating that DP-rcCFSFDP algorithm clusters with higher effectiveness.

In general, DP-rcCFSFDP reduces the impact of Laplacian noise on clustering compared with DP-CFSFDP and achieves a better balance between clustering effectiveness and privacy preserving.

5 Conclusion

In this paper, a density peak clustering algorithm based on differential privacy preserving (DP-CFSFDP) is proposed to protect private data. Meanwhile, an improved DP-CFSFDP algorithm with reachable-centers (DP-rcCFSFDP) is proposed for the poor performance on data with uniform distribution and the bad clustering with Laplacian noise by CFSFDP. The experiments show that the improved algorithm can meet the requirement of privacy preserving while ensuring the effectiveness of clustering. In the future, we are going to optimize the allocation of input parameters and privacy budget in DP-rcCFSFDP, and further improve the clustering performance.

References

1. Sweeney, L.: k-anonymity: a model for protecting privacy. Int. J. Uncertain. Fuzziness Knowl. Based Syst. 10(5), 557–570 (2002).
2. Karakasidis, A., Verykios, V.: Reference table based k-anonymous private blocking. In: Proceedings of the 27th Annual ACM Symposium on Applied Computing, pp. 859–864. ACM (2012).
3. Machanavajjhala, A., Gehrke, J., Kifer, D., Venkitasubramaniam, M.: l-diversity: Privacy beyond k-anonymity. In: 22nd IEEE International Conference on Data Engineering, pp. 24-24. IEEE (2006).
4. Li, N., Li, T., Venkatasubramanian, S.: t-closeness: Privacy beyond k-anonymity and l-diversity. In: IEEE 23rd International Conference on Data Engineering, pp. 106–115. IEEE (2007).
5. Dwork, C.: Differential privacy. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4052, pp. 1–12. Springer, Heidelberg (2006).
6. Dwork, C., Roth, A.: The algorithmic foundations of differential privacy. Found. Trends Theor. Comput. Sci. 9(3–4), 211–407 (2014).
7. Dwork, C., McSherry, F., Nissim, K., Smith, A.: Calibrating noise to sensitivity in private data analysis. In: TCC, pp. 265–284 (2006).
8. Blum, A., Dwork, C., McSherry, F., Nissim, K.: Practical privacy: the SuLQ framework. In: PODS, pp. 128–138. ACM (2005).
9. Wu, W., Huang, H.: A DP-DBScan clustering algorithm based on differential privacy preserving. Computer Engineering and Science 37(4), 830-834 (2015).
10. Nissim, K., Raskhodnikova, S., Smith, A.: Smooth sensitivity and sampling in private data analysis. In: STOC, pp. 75–84. ACM (2007).
11. Ren, J., Xiong, J., Yao, Z., Ma, R., Lin, M.: DPLK-means: A novel Differential Privacy K-means Mechanism. In: DSC 2017: IEEE Second International Conference on Data Science in Cyberspace, pp. 133-139. IEEE (2017).
12. Dwork, C. A firm foundation for private data analysis. Communications of the ACM, 54(1), 86–95 (2011).
13. Wang, H., Ge, L., Wang, S., et al.: Improvement of differential privacy protection algorithm based on OPTICS clustering. Journal of Computer Applications 38(1), 73-78 (2018).
14. Chen, L., Yu, T., Chirkova, R.: Wavecluster with differential privacy. In: CIKM, pp. 1011-1020. ACM (2015).

15. McSherry, F., Talwar, K.: Mechanism design via differential privacy. In: Proceedings of 48th Annual IEEE Symposium on Foundations of Computer Science, Providence, RI, pp. 94–103 (2007).
16. Dwork, C.: Differential privacy: A survey of results. In: Agrawal, M., Du, D.-Z., Duan, Z., Li, A. (eds.) TAMC 2008. LNCS, vol. 4978, pp. 1–19. Springer, Heidelberg (2008).
17. Dwork, C., Lei, J.: Differential Privacy and Robust Statistics. In: STOC, pp. 371–380 (2009).
18. Dwork, C., Naor, M., Reingold, O., Rothblum, G.N., Vadhan, S.: On the complexity of differentially private data release: efficient algorithms and hardness results. In: STOC 2009: Proceedings of the 41st Annual ACM Symposium on Theory of Computing, pp. 381–390. ACM, New York (2009).
19. Dwork, C.: The differential privacy frontier. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 496–502. Springer, Heidelberg (2009).
20. Rodriguez, A., Laio, A.: Clustering by fast search and find of density peaks. Science 344(6191), 1492–1496 (2014).
21. Ester, M., Kriegel, H., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the second ACM SIGKDD international conference on knowledge discovery and data mining, pp. 226–231 (1996).
22. Clustering datasets, http://cs.joensuu.fi/sipu/datasets/.
23. UCI Machine Learning Repository, https://archive.ics.uci.edu/ml/datasets.html.
24. Chinchor N.: MUC-4 Evaluation Metrics. In: Proc. of the Fourth Message Understanding Conference, pp. 22–29 (1992).
25. Hubert, L., Arabie, P.: Comparing partitions. Journal of Classification 2(1), 193-218 (1985).
26. Gates, A., Ahn, Y.: The impact of random models on clustering similarity. J. Mach. Learn. Res. 18(1), 3049-3076 (2017).
27. Zhang, Y., Wei, J., Zhang, X., et al.: A Two-Phase Algorithm for Generating Synthetic Graph Under Local Differential Privacy. In: ICCNS 2018: Proceedings of the 8th International Conference on Communication and Network Security, pp. 84-89. ACM, New York (2018).
28. André, L., Brito, F., et al.: DiPCoDing: A Differentially Private Approach for Correlated Data with Clustering. In: IDEAS 2017: Proceedings of the 21st International Database Engineering & Applications Symposium, pp. 291-297. ACM (2017).
29. Huang, Z., Liu, J.: Optimal Differentially Private Algorithms for k-Means Clustering. In: Proceedings of the 37th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, pp. 395-408. ACM, New York (2018).