A *k*-Anonymity Clustering Method for Effective Data Privacy Preservation

Chuang-Cheng Chiu and Chieh-Yuan Tsai

Department of Industrial Engineering and Management, Yuan Ze University, Taiwan
cytsai@saturn.yzu.edu.tw

Abstract. Data privacy preservation has drawn considerable interests in data mining research recently. The *k*-anonymity model is a simple and practical approach for data privacy preservation. This paper proposes a novel clustering method for conducting the *k*-anonymity model effectively. In the proposed clustering method, feature weights are automatically adjusted so that the information distortion can be reduced. A set of experiments show that the proposed method keeps the benefit of scalability and computational efficiency when comparing to other popular clustering algorithms.

Keywords: Data privacy preservation, *k*-Anonymity, Clustering, C-means clustering algorithm, Feature weighting.

1 Introduction

Rapid advances in database technologies enabled organizations to accumulate vast amounts of data in recent years. Data mining has been a common methodology to retrieve and discover useful knowledge from these growing data [1]. In many industrial applications, many personal details and sensitive information are contained in these data such as financial transactions, telephone communication traffic, health care records, and so on. The knowledge extracted from these data may unwittingly uncover personal sensitive information. Therefore, before conducting data mining, these data must be protected through some privacy-preserving techniques. This makes privacy-preserving becomes an important issue in data mining fields in recent years [2, 3].

The *k*-anonymity model, proposed by Sweeney [4], is a simple and practical privacy-preserving approach and is extensively studied recently [5, 6, 7]. The *k*-anonymity model ensures that each record in the table is identical to at least (*k*-1) other records with respect to the privacy-related features. Therefore, no privacy-related information can be inferred from the *k*-anonymity protected table during a data mining process. For example, patient diagnosis records without conducting the *k*-anonymity model is shown in Fig. 1(a) [8]. It is clear that a diagnosis classifier can be developed using these data to predict patient's illness based on features of Zip, Gender, and Age. If the hospital simply publishes the table to other organizations for classifier development, the organizations might extract patients' disease history by joining this table with other tables. Conversely, if the *k*-anonymity model is conducted for these data, data values in features Zip, Gender, and Age might been

R. Alhajj et al. (Eds.): ADMA 2007, LNAI 4632, pp. 89–99, 2007.

generalized as capsule values shown in Fig 1(b). For each patient in the table, we can find that at least two patients have the same Zip, Gender, and Age feature values with him/her. Therefore, when the hospital publishes such a *k*-anonymity protected table to other organizations, the organizations still develops an illness-diagnosing classifier from this table similarly. Importantly, the organizations can not uncover additional information from each patient's generalized feature values. The purpose of data privacy preservation is then achieved.

Zip	Gender	Age	Diagnosis
47918	Male	35	Cancer
47906	Male	33	HIV+
47918	Male	36	Flu
47916	Female	39	Obesity
47907	Male	33	Cancer
47906	Female	33	Flu

(a) Patient diagnosis records in a hospital

Zip	Gender	Age	Diagnosis
4791*	Person	[35-39]	Cancer
4790*	Person	[30-34]	HIV+
4791*	Person	[35-39]	Flu
4791*	Person	[35-39]	Obesity
4790*	Person	[30-34]	Cancer
4790*	Person	[30-34]	Flu

(b) The *k*-anonymity protected table of (a) when $k = 3$

Fig. 1. An example of data privacy preservation using the *k*-anonymity model

In the *k*-anonymity model, the *quasi-identifier feature set* consists of features in a table that potentially reveals private information, possibly by joining with other tables. In addition, the *sensitive feature* is a feature serves as the class label of each record. As shown in Fig. 1(b), the set of three features {Zip, Gender, Age} is the quasi-identifier feature set, while the feature {Diagnosis} is the sensitive feature. For each record in this table, its feature values in the quasi-identifier feature set are generalized as capsule feature values, while its value of sensitive feature are not generalized. Through generalization, an *equivalence class* is the set composed of records in the table which has the same values on all features in the quasi-identifier feature set. The 1st, 3rd and 4th records in Fig. 1(b) are assembled to form one equivalence class, while the 2nd, 5th and 6th records are assembled to form another equivalence class. The number of records in each equivalence class must be not less than *k*, which is called as the *k*-anonymity requirement. The value of *k* is specified by users according to the purpose of their applications. The records in Fig. 1(b) satisfy 3-anonymity requirement since the numbers of records in its two equivalence classes are both equal to three.

To ensure data mining performance, usability should be taken into account when constructing the *k*-anonymity protected table [8]. The less the information distortion in the *k*-anonymity protected table makes, the larger the table usability is. Therefore, a *k*-anonymity model must minimize the information distortion from its original table. Unfortunately, the computational complexity of finding an optimal solution for *k*-anonymity model has been shown to be NP-hard [9]. In recent years, many clustering techniques based on heuristic scheme have been developed to conduct the *k*-anonymity protected table [5, 7, 8, 9, 10, 11]. Clustering [12] aims at grouping a set of objects into clusters so that objects in a cluster are similar to each other and are different from objects in other clusters. In the *k*-anonymity protected table, if the records that will be assembled as an equivalence class are more similar to each other, it retrenches the more information distortion for generalizing the equivalence class. That is the reason why the *k*-anonymity model can be addressed from the viewpoint

of clustering. Among various types of clustering methods, hierarchical clustering methods are frequently used to conduct the *k*-anonymity protected table [5, 8, 9, 10]. Although their efforts are admirable in the issue, their computational efficiency may degenerate when the amount of records increases. Furthermore, how to define a proper similarity/dissimilarity measure between two equivalence classes is another challenge when using hierarchical clustering methods.

A novel clustering method to construct the *k*-anonymity protected table is proposed in this paper. In the proposed method, a Weighted Feature C-means clustering algorithm (WF-C-means) is proposed to partition all records into equivalence classes. For enhancing clustering quality, WF-C-means adaptively adjusts the weight of each quasi-identifier feature based on the importance of the feature to clustering quality. The operational procedure in WF-C-means is similar to the C-means algorithm [13] which has good scalability for large data, so that the computational efficiency of WF-C-means is practicable in practice. After completing the clustering, a class-merging mechanism merges equivalence classes to make sure that all equivalence classes satisfy the *k*-anonymity requirement. All records in each equivalence class are generalized to be the same with the class center in the class. Through our experiments, the proposed clustering method outperforms existing methods in terms of information distortion measure and computational efficiency.

2 The Proposed *k*-Anonymity Clustering Method

The core of the proposed clustering method for constructing the *k*-anonymity protected table consists of a Weighted Feature C-means clustering algorithm (WF-C-means) and a class-merging mechanism, and is introduced respectively in detail as follows.

2.1 A Weighted Feature C-Means Clustering Algorithm

Let a table $\mathbf{T}=\{\mathbf{r}_1,\ldots,\mathbf{r}_m,\ldots,\mathbf{r}_M\}$ include M records and a quasi-identifier feature set $\mathbf{F}=\{\mathbf{f}_1,\ldots,\mathbf{f}_n,\ldots,\mathbf{f}_N\}$ comprise N features. A record $\mathbf{r}_m=(r_{m1},\ldots,r_{mn},\ldots,r_{mN})$ is composed of N quasi-identifier feature values where r_{mn} is the value of the nth quasi-identifier feature $\mathbf{f}_n$ in the mth record $\mathbf{r}_m$. Noted that the sensitive feature values can not be generalized as capsule values so that the sensitive feature is not involved in the record $\mathbf{r}_m$.

The development of the proposed WF-C-means is derived from the C-means clustering algorithm [13]. WF-C-means aims at partitioning all M records in the table $\mathbf{T}$ into C equivalence classes. The number of equivalence classes, C, depends on the value of k specified in the *k*-anonymity model, which is shown as Equation (1):

$$C = M \setminus k \tag{1}$$

where "\" is the integer division operator. For example, in the 3-anonymity model a table of 100 records will be divided into 33 equivalence classes (100\3=33). Let $\mathbf{C}=\{\mathbf{C}_1,\ldots,\mathbf{C}_i,\ldots,\mathbf{C}_C\}$ be the set of the C equivalence classes and $\mathbf{A}=\{\mathbf{a}_1,\ldots,\mathbf{a}_i,\ldots,\mathbf{a}_C\}$ be the set of the C class centers in $\mathbf{C}$ where $\mathbf{a}_i=(a_{i1},\ldots,a_{in},\ldots,a_{iN})$ is the class center of the ith equivalence class $\mathbf{C}_i$ and a_{in} is the value of the nth quasi-identifier feature $\mathbf{f}_n$ in the ith class center $\mathbf{a}_i$.

Accordingly, the dissimilarity between a record $\mathbf{r}_m$ and a class center $\mathbf{a}_i$, termed as diss($\mathbf{r}_m$,$\mathbf{a}_i$), can be defined as:

$$\text{diss}(\mathbf{r}_m, \mathbf{a}_i) = \sum_{n=1}^{N} w_n \times \text{diss}(r_{mn}, a_{in}) \tag{2}$$

where $w_n \in \mathbf{w}$ is the weight of the quasi-identifier feature $\mathbf{f}_n$ and $\mathbf{w}=\{w_1,\ldots,w_n,\ldots,w_N\}$ is the set of the N weights associated with N quasi-identifier features in $\mathbf{F}$, $\sum_{n=1}^{N} w_n = 1$, $0 \le w_n \le 1$. Furthermore, diss(r_{mn},a_{in}) is the dissimilarity between $\mathbf{r}_m$ and $\mathbf{a}_i$ in terms of the nth quasi-identifier feature. Numerical features and categorical features have their respective formulas to evaluate the value of diss(r_{mn},a_{in}). In this paper we assume all quasi-identifier features in $\mathbf{F}$ are numerical since we emphasize the introduction of this clustering algorithm. Therefore, the evaluation formula of diss(r_{mn},a_{in}) can be defined as Equation (3). Noted that the details about the dissimilarity evaluation for categorical features can be referred to [10].

$$\text{diss}(r_{mn}, a_{in}) = (r_{mn} - a_{in})^2 \tag{3}$$

The objective of WF-C-means, equivalent to C-means, is to minimize the sum of the dissimilarities between all M records to their corresponding class centers, which can be expressed as follows:

$$\textit{Minimize}\ \ S(\mathbf{U}, \mathbf{A}, \mathbf{w}) = \sum_{m=1}^{M}\sum_{i=1}^{C} u_{mi} \times \text{diss}(\mathbf{r}_m, \mathbf{a}_i) = \sum_{m=1}^{M}\sum_{i=1}^{C}\sum_{n=1}^{N} u_{mi} \times w_n \times \text{diss}(r_{mn}, a_{in}) \tag{4}$$

subject to

$$\begin{cases} \sum_{i=1}^{C} u_{mi} = 1 \\ u_{mi} \in \{1, 0\} \quad \text{for } m = 1,2,\cdots,M;\ i = 1,2,\cdots,C;\ n = 1,2,\cdots,N \\ \sum_{n=1}^{N} w_n = 1 \\ w_n \ge 0 \end{cases} \tag{5}$$

where $\mathbf{U}$ is a matrix of size $M \times C$ that stores the record-class memberships and $u_{mi} \in \{1, 0\}$ is an element in $\mathbf{U}$ that represents the membership of the record $\mathbf{r}_m$ with the ith cluster $\mathbf{C}_i$. If u_{mi}=1, $\mathbf{r}_m$ belongs to $\mathbf{C}_i$. If u_{mi}=0, by contrast, $\mathbf{r}_m$ does not belong to $\mathbf{C}_i$.

The WF-C-means algorithm solves the described optimization problem by iteratively solving the following three reduced problems until all elements in the record-class membership matrix $\mathbf{U}$ remain the same without being changed.

1. Problem P_1: Fix $\mathbf{A} = \hat{\mathbf{A}}$ and $\mathbf{w} = \hat{\mathbf{w}}$ to solve the reduced problem $S(\mathbf{U}, \hat{\mathbf{A}}, \hat{\mathbf{w}})$.
2. Problem P_2: Fix $\mathbf{U} = \hat{\mathbf{U}}$ and $\mathbf{w} = \hat{\mathbf{w}}$ to solve the reduced problem $S(\hat{\mathbf{U}}, \mathbf{A}, \hat{\mathbf{w}})$.
3. Problem P_3: Fix $\mathbf{A} = \hat{\mathbf{A}}$ and $\mathbf{U} = \hat{\mathbf{U}}$ to solve the reduced problem $S(\hat{\mathbf{U}}, \hat{\mathbf{A}}, \mathbf{w})$.

The purpose of solving P_1 is to assign a record to an equivalence class in which the class center is closest to the record. Therefore, the procedure for solving P_1 is called a record-assignment procedure and is expressed in Equation (6):

$$\begin{cases} u_{mi} = 1, \text{ if } \sum_{n=1}^{N} w_n \times \text{diss}(r_{mn}, a_{in}) \le \sum_{n=1}^{N} w_n \times \text{diss}(r_{mn}, a_{jn}) \\ u_{mi} = 0, \text{Otherwise} \end{cases} \quad \text{for } 1 \le i, j \le C, \ j \ne i \tag{6}$$

Accordingly, the purpose of solving the problem P_2 is to update all K class centers in the C classes respectively. Therefore, the procedure for solving P_2 is called as a center-updating procedure and is expressed in Equation (7):

$$a_{in} = \sum_{m=1}^{M} u_{mi} \times r_{mn} \Big/ \sum_{m=1}^{M} u_{mi} \quad \text{for } i = 1,2,\cdots,C; \ n = 1,2,\cdots,N \tag{7}$$

The difference between WF-C-Means and C-means is that WF-C-Means further solves the weight-adjusting problem P_3 but C-means does not. The weight of a quasi-identifier feature should reflect the importance of the feature to the clustering quality, measured by how the feature can achieve the clustering objective function of minimizing the separations within clusters and maximizing the separations between clusters simultaneously. If a feature is important, increasing its feature weight should make the clustering objective function be easily achieved. Therefore, the goal of sub-problem P_3 is to

$$Maximize \ V(\hat{\mathbf{U}}, \hat{\mathbf{A}}, \mathbf{w}, \hat{g}) = \frac{S'(\hat{\mathbf{A}}, \mathbf{w}, \hat{g})}{S(\hat{\mathbf{U}}, \hat{\mathbf{A}}, \mathbf{w})} = \frac{\sum_{n=1}^{N} \left[w_n \times \left(\sum_{i=1}^{C} \| \mathbf{C}_i \| \times \text{diss}(a_{in}, g_n) \right) \right]}{\sum_{n=1}^{N} \left[w_n \times \left(\sum_{m=1}^{M} \sum_{i=1}^{C} u_{mi} \times \text{diss}(r_{mn}, a_{in}) \right) \right]} \tag{8}$$

subject to

$$\begin{cases} \sum_{n=1}^{N} w_n = 1 \\ w_n \ge 0 \end{cases} \quad \text{for } n = 1,2,\cdots,N \tag{9}$$

where $S(\hat{\mathbf{U}}, \hat{\mathbf{A}}, \mathbf{w})$ is the sum of all separations within clusters and $S'(\hat{\mathbf{A}}, \mathbf{w}, \hat{g})$ is the sum of all separations between clusters. Noted that $g=(g_1,\ldots,g_n,\ldots,g_N)$ is the global center of all M records in the table $\mathbf{T}$, and its nth feature value, g_n, can be evaluated by $g_n = \sum_{m=1}^{M} r_{mn} \big/ M$. In addition, $\| \mathbf{C}_i \|$ represents the number of records in the ith cluster $\mathbf{C}_i$ such that $\sum_{i=1}^{C} \| \mathbf{C}_i \| = M$.

Let $e_n = \sum_{m=1}^{M} \sum_{i=1}^{C} u_{mi} \times \text{diss}(r_{mn}, a_{in})$ be the sum of separations within clusters in terms of $\mathbf{f}_n$ and $f_n = \sum_{i=1}^{C} \| \mathbf{C}_i \| \times d(a_{in}, g_n)$ be the sum of separations between clusters in terms of $\mathbf{f}_n$. Accordingly, Equation (8) can be rewritten as:

$$Maximize \ V(\hat{\mathbf{U}}, \hat{\mathbf{A}}, \mathbf{w}, \hat{g}) = \frac{\sum_{n=1}^{N} w_n \times f_n}{\sum_{n=1}^{N} w_n \times e_n} \tag{10}$$

subject to

$$\begin{cases} \sum_{n=1}^{N} w_n = 1 \\ w_n \geq 0 \end{cases} \quad \text{for } n = 1,2,\cdots,N \tag{11}$$

This research proposes an adaptive weight-adjusting principle to derive **w** from Equation (10). Let $\{w_1^{(s)},\cdots,w_n^{(s)},\cdots,w_N^{(s)}\}$ be the set of the N feature weights at the sth iteration (i.e. current iteration) in WF-C-Means. Each feature weight $w_n^{(s+1)}$ for n=1, 2, …,N at the (s+1)th iteration (i.e. next iteration) in WF-C-Means will be adjusted by adding an adjustment margin Δw_n , which is shown as Equation (12).

$$w_n^{(s+1)} = w_n^{(s)} + \Delta w_n \quad \text{for } n = 1,2,\cdots,N \tag{12}$$

The adjustment margin Δw_n for feature $\mathbf{f}_n$ is evaluated based on how important the feature contributes to clustering quality. From Equation (10), we know that feature $\mathbf{f}_n$ possessing a high (f_n/e_n) value should have a high weight value. Therefore, adjustment margin Δw_n can be derived according to its (f_n/e_n) value using the following equation:

$$\Delta w_n = \frac{f_n/e_n}{\sum_{n=1}^{N}(f_n/e_n)} \quad \text{for } n = 1,2,\cdots,N \tag{13}$$

Accordingly, an adjusted feature weight $w_n^{(s+1)}$ can be rewritten as:

$$w_n^{(s+1)} = w_n^{(s)} + \frac{f_n/e_n}{\sum_{n=1}^{N}(f_n/e_n)} \quad \text{for } n = 1,2,\cdots,N \tag{14}$$

In addition, the adjusted weight in Equation (14) need to be normalized as the value between 0 to 1 to satisfy the constraint of $\sum_{n=1}^{N} w_n^{(s+1)} = 1$. Therefore, a simple normalization function $f(t_n)$ defined in Equation (15) is used in this paper.

$$f(t_n) = \frac{t_n}{\sum_{n=1}^{N} t_n} \quad \text{for } n = 1,2,\cdots,N \tag{15}$$

Through the normalization function, each adjusted feature weight $w_n^{(s+1)}$ can be derived as:

$$w_n^{(s+1)} = f(w_n^{(s+1)}) = \frac{w_n^{(s)} + \frac{f_n/e_n}{\sum_{n=1}^{N}(f_n/e_n)}}{\sum_{n=1}^{N} w_n^{(s)} + \sum_{n=1}^{N}\left(\frac{f_n/e_n}{\sum_{n=1}^{N}(f_n/e_n)}\right)} \quad \text{for } n = 1,2,\cdots,N \tag{16}$$

With Equation (16), the adjusted feature weights can be derived in the weight-adjusting procedure and feed back to the beginning of the record-assignment procedure in the WF-C-means algorithm for the next iteration. The pseudo-code of the WF-C-means algorithm is summarized in Fig. 2.

Input: a table **T** contains *M* records in which each record has *N* quasi-identifier features;
the value of *k* in the *k*-anonymity model.
1: Calculate the number of equivalence classes *C* using Equation (1).
2: Randomly select *C* records from **T** as the class centers of the *C* equivalence classes.
3: Let the weight of each quasi-identifier feature be (1/*N*).
4: **Repeat**
5: Form *C* equivalence classes by assigning each record to its closest class center using Equation (6).
6: Update the class center in each equivalence class using Equation (7).
7: Adjust the feature weight of each quasi-identifier feature using Equation (16).
8: **Until** all elements in the record-class membership matrix do not change

Fig. 2. The pseudo-code of the WF-C-means algorithm

2.2 A Class-Merging Mechanism

After executing the proposed WF-C-means algorithm, a few equivalence classes may violate the *k*-anonymity requirement because they are possibly located at the purlieus of data distribution or even they are outliers. Assume there are *P* illegal equivalence classes violating the *k*-anonymity requirement among all *C* equivalence classes, so that other $(C-P)$ equivalence classes are legal. In the proposed method, a class-merging mechanism is developed to eliminate the illegal equivalence classes by means of merging them with legal equivalence classes.

Let the distance between two equivalence classes $\mathbf{C}_i$ and $\mathbf{C}_j$ be defined as the dissimilarity between their class centers $\mathbf{a}_i$ and $\mathbf{a}_j$, which is expressed as:

$$\text{distance}(\mathbf{C}_i, \mathbf{C}_j) = \text{diss}(\mathbf{a}_i, \mathbf{a}_j) = \sum_{n=1}^{N} w_n \times \text{diss}(a_{in}, a_{jn}) \tag{17}$$

where the two class centers $\mathbf{a}_i$ and $\mathbf{a}_j$, and w_n for n=1,2,…,N are known after executing the WF-C-means algorithm. When merging x equivalence classes, the class center of a new equivalence class, termed as $\mathbf{a}^{\text{new}} = (a_1^{\text{new}}, \ldots, a_n^{\text{new}}, \ldots, a_N^{\text{new}})$, can be defined as Equation (18):

$$a_n^{\text{new}} = \frac{\sum_{i=1}^{x} (a_{in} \times \| \mathbf{C}_i \|)}{\| \mathbf{C}_i \|} \quad \text{for } n = 1, \ldots, N \tag{18}$$

where $\|\mathbf{C}_i\|$ is the number of records in the equivalence class $\mathbf{C}_i$. Noted that the number of records in the new equivalence class equals to $\sum_{i=1}^{x} \| \mathbf{C}_i \|$.

For an illegal equivalence class, its merging target is the legal equivalence class with closest distance evaluated by Equation (17). For a legal equivalence class, on the other hand, it may receive the merging requests from several illegal equivalence classes so that it will be merged with these illegal equivalence classes simultaneously. The class center of the new equivalence class can be found easily by Equation (18). The pseudo-code of the class-merging procedure in the proposed mechanism is summarized in Fig. 3. After performing the class-merging mechanism, all records in each equivalence class $\mathbf{C}_i$ are generalized to be the same with the class center $\mathbf{a}_i$ of $\mathbf{C}_i$.

```
Input: P illegal equivalence classes and (C - P) legal equivalence classes which are generated
       from WF-C-means
1:   For each illegal equivalence class
2:     Calculate the distances with the (C - P) legal classes respectively using Equation (17).
3:     Select and mark the nearest legal class with it.
4:   For each legal equivalence class
5:     If the class has been marked by any illegal equivalence class
6:       Collect the illegal equivalence classes which have done a mark on it.
7:       Merge these collected illegal equivalence classes with it as a new equivalence class.
```

Fig. 3. The pseudo-code of the class-merging procedure in the proposed mechanism

3 Experiments

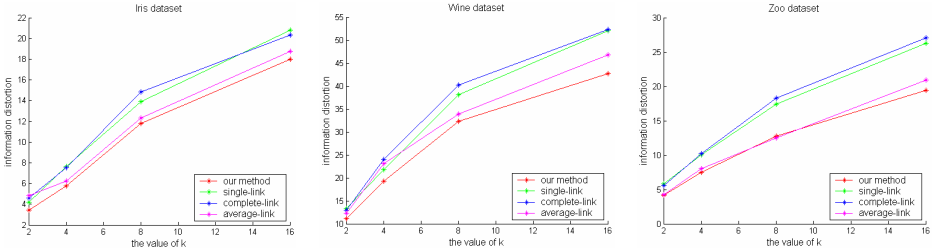
To show performance of the proposed *k*-anonymity clustering method, a series of experiments using Iris, Wine, and Zoo datasets from UCI machine learning repository [15] are conducted. For each dataset, its original predictive features are all in the quasi-identifier feature set while its class-label feature is the sensitive feature. Since hierarchical clustering methods have been adopted most frequently in previous studies, the experiment result of the proposed method is compared with the results of three common hierarchical clustering methods. They are single-link, complete-link, and average-link clustering methods [16]. In a hierarchical clustering method, all records are initially considered as independent equivalence classes and are merged progressively until the number of records in each equivalence class is not less than *k*.

3.1 Information Distortion

The amount of information distortion can be evaluated from the difference between the original table and the *k*-anonymity protected table. For each record in the *k*-anonymity protected table, its feature values in the quasi-identifier feature set are generalized as the feature values of the equivalence class center which is closet to it. Therefore, the amount of information distortion of a *k*-anonymity protected table can be calculated using Equation (4). The less the amount of information distortion, the larger the usability of the *k*-anonymity protected table is.

When using the hierarchical clustering methods, all the feature weights in Equation (4) are set as $1/N$ uniformly and the class center of an equivalence class is considered as the mean of all records in the equivalence class. In addition, the parameter, *k*, is tested using 2, 4, 8, 16, respectively, for each dataset. The plots of the amounts of information distortion with respect to *k* using the three datasets for the four clustering methods are illustrated in Fig. 4.

From Fig. 4, it is clear that the amount of information distortion increases as *k* grows no matter which clustering method is used to develop the *k*-anonymity model. Among these four clustering methods, the proposed *k*-anonymity clustering method is the best one to restrain information distortion for all *k* values and datasets.



(a) Iris dataset (b) Wine dataset (c) Zoo dataset

Fig. 4. The plots of the information distortion with respect to *k* using the three datasets for the four clustering methods

3.2 Classification Error Rate

In the study, we assume that the one nearest neighbor (1NN) classification technique is used to classify unknown data based on the *k*-anonymity protected table in following data mining tasks. In the classification task, each record in the original table serves as a testing sample to measure the classification error rate. Therefore, the less the classification error rate, the larger the usability of the *k*-anonymity protected table is. Table 1 shows classification error rates for the four clustering methods using the three datasets when $k = 3$. It is clear that the performance of the proposed method is superior to other three clustering methods in Wine and Zoo datasets, while only slightly inferior to the average-link hierarchical clustering method in the Iris dataset. For our method, in addition, the classification error rate increases lightly using the *k*-anonymity protected table when comparing to the one using the original table. However, the data privacy in the *k*-anonymity protected table can be strongly preserved.

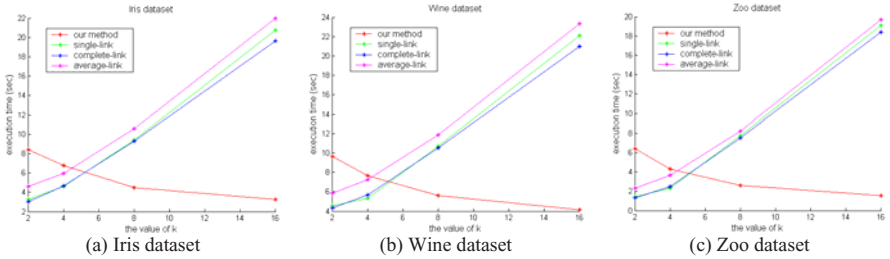
Table 1. The classification error rates using the three datasets for the four clustering methods

Dataset	Original table	*k*-anonymity Protected table			
		Our method	Hierarchical clustering method		
			single-link	complete-link	average-link
Iris	4.47%	6.67%	8.67%	9.33%	**6.00%**
Wine	15.73%	**17.42%**	22.47%	21.91%	19.66%
Zoo	56.25%	**59.38%**	65.63%	62.50%	62.50%

(Noted that the value of *k* is set as 3 in the *k*-anonymity model.)

3.3 Computational Efficiency

In this section, the computational efficiency of constructing the *k*-anonymity protected table using the four clustering methods is evaluated by measuring their execution time. The experiment environment is set identically with the one in Section 3.1. All experiments are implemented with Excel VBA programming language, and run on an AMD K7 2.5G personal computer with 512 MB memory. The plots of the execution time with respect to *k* for the three datasets using the four clustering algorithms are illustrated in Fig. 5.



(a) Iris dataset (b) Wine dataset (c) Zoo dataset

Fig. 5. The plots of the execution time with respect to k for the three datasets using the four clustering algorithms

The execution time using the proposed clustering method decreases as k increases, which is totally different to the trends using the three hierarchal clustering methods. Moreover, the execution time using proposed clustering method is less than 10 seconds for all k value settings and datasets. The computational complexity of the proposed clustering method is $O(M \times C) \cong O(M \times (M/k)) = O(M^2/k)$, while the computational complexity of a hierarchical clustering method equals to $O(M^2 \log M)$. It is obvious that the proposed method is superior to the hierarchical clustering method in terms of computational efficiency.

4 Conclusion

In this paper we propose a novel C-means type clustering method for the k-anonymity model, which is distinct from the typical hierarchical clustering methods. For restraining information distortion in the k-anonymity protected table, the proposed method adaptively adjusts the weight of each quasi-identifier feature based on the importance of the feature to clustering quality. The experiment results in Section 3.1 and Section 3.2 also confirms that the proposed method enables the k-anonymity protected table restrain its information distortion. In addition, the experiment result in Section 3.3 indicates that the computational efficiency of the proposed clustering method is superior to the hierarchical clustering method for the k-anonymity model.

In this paper only quasi-identifier features with numerical values are considered. However, quasi-identifier features with categorical values are also common in practice. In the future, we will focus on developing a dissimilarity-evaluating approach which takes different types of feature values into account simultaneously.

References

1. Tan, P.N., Steinbach, M., Kumar, V.: Introduction to Data Mining, pp. 487–559. Addison-Wesley, Boston (2005)
2. Agrawal, R., Srikant, R.: Privacy-Preserving Data Mining. SIGMOD Record 29, 439–450 (2000)
3. Lindell, Y., Pinkas, B.: Privacy Preserving Data Mining. Journal of Cryptology 15, 177–206 (2003)

4. Sweeney, L.: k-Anonymity: A Model for Protecting Privacy. International Journal of Uncertainty, Fuzziness and Knowlege-Based Systems 10, 557–570 (2002)
5. Domingo-Ferrer, J., Torra, V.: Ordinal, Continuous and Heterogeneous k-Anonymity through Microaggregation. Data Mining and Knowledge Discovery 11, 195–212 (2005)
6. LeFevre, K., DeWitt, D.J., Ramakrishnan, R.: Incognito: Efficient Full-Domain k-Anonymity. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 49–60 (2005)
7. Li, X.-B., Sarkar, S.: A Tree-Based Data Perturbation Approach for Privacy-Preserving Data Mining. IEEE Transactions on Knowledge and Data Engineering 18, 1278–1283 (2006)
8. Byun, J.-W., Kamra, A., Bertino, E., Li, N.: Efficient k-Anonymization Using Clustering Techniques. To appear in the International Conference on Database Systems for Advanced Applications (2007)
9. Meyerson, A., Williams, R.: On the Complexity of Optimal k-Anonymity. In: Proceedings of the 18th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, pp. 223–228 (2004)
10. Jiuyong, L., Wong, R.C.-W., Fu, A.W.-C., Jian, P.: Achieving k-Anonymity by Clustering in Attribute Hierarchical Structures. In: Tjoa, A.M., Trujillo, J. (eds.) DaWaK 2006. LNCS, vol. 4081, pp. 405–416. Springer, Heidelberg (2006)
11. Aggarwal, C.C.: On k-Anonymity and the Curse of Dimensionality. In: Proceedings of the 31st International Conference on Very Large Data Bases, pp. 901–909 (2005)
12. Jain, A., Dube, R.: Algorithms for Clustering Data. Prentice Hall, New Jersey (1988)
13. McQueen, J.: Some Methods for Classification and Analysis of Multivariate Observations. In: Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, pp. 281–297 (1967)
14. Hillier, F.S., Lieberman, G.J.: Introduction to Operation Research. McGraw-Hill, New York (2001)
15. Newman, D.J., Hettich, S., Blake, C.L., Merz, C.J.: UCI Repository of Machine Learning Databases (1998), available at http://www.ics.uci.edu/~mlearn/MLSummary.html
16. Jain, A.K., Murty, M.N., Flynn, P.J.: Data Clustering: A Review. ACM Computer Survey 31, 264–323 (1999)