Journal of Ambient Intelligence and Humanized Computing
https://doi.org/10.1007/s12652-019-01468-y

ORIGINAL RESEARCH

Data anonymization with imprecise rules and its performance evaluations

Masahiro Inuiguchi[1] · Hiroki Ichida[1] · Vicenç Torra[2]

Received: 11 February 2019 / Accepted: 20 August 2019

Abstract

Privacy protection is absolutely imperative for data releases when the utilization of public data and big data is getting popular. In this paper, data anonymization methods using rough set-based rule induction are investigated. It has been shown that many rules with imprecise conclusions can improve the classification accuracy of the rule-based classifier. Data anonymization methods utilizing rules with imprecise conclusions are proposed. The data tables anonymized by one of the proposed methods can preserve the classification accuracy of the rules induced from them. The proposed methods as well as conventional data anonymization methods are compared from two viewpoints: the classification accuracy of rules induced from the anonymized data table and the preservation of data anonymity. The results show the usefulness of the proposed methods.

Keywords Rule induction · Imprecise rules · Privacy protection · Data anonymization

1 Introduction

As information systems, sensor technologies and data storage equipments are developed and spread into many fields, various kind of data are stored and utilized for discovering some new knowledge. This leads to a high requirement for research and development of machine learning, data mining and knowledge discovery. For scientific developments by knowledge discovery from data, data publication and exchange are getting popular. This makes privacy protection absolutely imperative for data releases. Many privacy preservation techniques (Fung et al. 2010; Torra 2017) have been proposed. Applying privacy protection techniques, the quality as well as the usefulness of the original data are deteriorated. Balancing the privacy protection ability and the data quality preservation is a difficult issue.

Data privacy has also been considered in data mining (Mendes and Vilela 2017; Abidi et al. 2019; Lekshmy and Rahiman 2019). Privacy protection has been considered extensively in the context of associate rule mining (Rizvi and Haritsa 2002; Evfimuievski et al. 2004). In this paper, we focus on methods for privacy protection when rough set approaches (Pawlak 1982) are applied to data mining. By means of rough sets, we find consistent objects and conflicting objects in data tables. Several methods for inducing decision rules from consistent objects have been proposed based on rough sets. In decision rule induction based on rough sets, minimal length rules are obtained from data tables called decision tables, in which attributes are divided into two kinds: condition attributes and decision attributes. Condition attributes are the attributes used to describe the profiles and characters of objects and used for describing the premises, while decision attributes are the attributes showing the results of classifications/evaluations and used for describing the conclusions. An induced decision rule is usually accurate so that all objects satisfying its premise take same decision attribute value specified in its conclusion. Although decision rule induction has been studied for a long time in rough set community, the aspects related to privacy have not yet been investigated in detail. We may found only a few studies (Ye et al. 2013; Inuiguchi et al. 2015; Inuiguchi and Washimi 2019) on data privacy using rough set approaches. In order to enhance the usability of anonymous data, attribute reduction based on rough sets is utilized in an anonymization approach (Ye et al. 2013). The concept of

✉ Masahiro Inuiguchi
inuiguti@sys.es.osaka-u.ac.jp

Vicenç Torra
vtorra@ieee.org

[1] Graduate School of Engineering Science, Osaka University, Toyonaka, Osaka 560-8531, Japan

[2] Hamilton Institute, Maynooth University, Maynooth, Kildare, Ireland

Published online: 17 September 2019

k-anonymity is introduced into rough set-based rule induction from decision tables with more than two decision attribute values (Inuiguchi et al. 2015). k-anonymous rules are the rules supported by at least k objects in the given table. To achieve the k-anonymity, imprecise rules (Inuiguchi et al. 2015) are utilized. The k-anonymous rules are utilized to anonymize a given decision table so as to preserve the usefulness of the decision table (Inuiguchi and Washimi 2019).

In this paper, we extend the study on the anonymization of decision tables by using the imprecise rules proposed in a conference paper (Inuiguchi and Washimi 2019). In this method, first k-anonymized decision rules are induced by using imprecise rules proposed by Inuiguchi et al. (2015). It is shown that the set of k-anonymized decision rules performs well in classification. The anonymized decision table called a k-common pattern table is produced by embedding several k-anonymous decision rules selected for each consistent object in the original table so as to specify its decision attribute value uniquely. We note that not all condition attribute values are specified by a k-anonymous decision rule. If a condition attribute value is not specified, the set of the attribute values of k objects supporting the rule is computed. By means of a numerical analysis, we observe a better privacy preservation in the k-common pattern tables than in the original decision table in the attack on the decision attribute with a few revealed condition attributes. Moreover, it was observed also that the deterioration of the quality of the imprecise rules induced from the table is not very high.

The privacy preservation and the quality are evaluated only by comparison with the original decision table. The advantage of k-common pattern tables in comparison with the conventional k-anonymous tables have not yet been clarified. Moreover, there are two conflicting ideas in building k-common pattern tables. One is the idea that a k-common pattern table with less difference from the original one is better. The other is the totally opposite idea that a k-common pattern table with more difference from the original one is better. The former idea comes from the quality/usefulness of the anonymized table while the latter idea comes from the privacy protection. Namely, there are some options in building k-common pattern tables. The performance differences of k-common pattern tables built from those different ideas have not yet examined. Attacks on condition attribute values have not yet been examined and the assumed attack is only based on a shallow analysis. The attack by a deep analysis was neither examined at all.

In this paper, four building methods for k-common pattern tables are investigated. In order to compare the conventional methods, we apply two anonymization methods based on Mondrian (LeFevre et al. 2005; Torra 2017). We compare the performances of four k-common pattern tables as well as two anonymization methods based on LeFevre et al. (2005) from the viewpoints of the privacy preservation and the quality/usefulness of the anonymized tables. The quality/usefulness of the anonymized tables is assessed by the classification accuracy of a classifier composed of imprecise rules induced from the anonymized tables. On the other hand, the privacy preservation is evaluated by the success ratios of decision and condition attribute attacks. In this evaluation, we apply a deep attack to a small decision table as well as the shallow attack.

This paper is organized as follows. In Sect. 2, we introduce the rough set approach and the induction of imprecise rules. The proposed data anonymization approach is described in Sect. 3. The procedures of the proposed anonymization methods are explained. In Sect. 4, methods and results of numerical experiments are described. Two kinds of experiments are executed: one is to examine the usefulness of the anonymized data tables and the other is to investigate the privacy protection abilities of the anonymized data tables. Some concluding remarks are given in Sect. 5.

2 Rough set approach and imprecise rule induction

Rough set approaches provide useful tools for analyzing decision tables. A decision table is a data table defined by a four-tuple $DT = \langle U, C \cup \{d\}, V, f \rangle$, where U is a finite set of objects, C is a finite set of condition attributes, d is a decision attribute, $V = \bigcup_{a \in C \cup \{d\}} V_a$ with attribute value set V_a of attribute $a \in C \cup \{d\}$ and $f : U \times C \cup \{d\} \to V$ is called an information function which is a total function. Condition attributes are attributes characterizing objects by their values. Namely, the profile of an object is represented by condition attribute values. Decision attribute shows a class to which the object belongs.

A very simple decision table is illustrated in Table 1. In Table 1, $U = \{o_1, o_2, o_3, o_4, o_5, o_6\}$, $C = \{\text{sex}, \text{occupation}, \text{domicile}\}$, $d = \text{evaluation}$, $V_{\text{sex}} = \{\text{male}, \text{female}\}$, $V_{\text{occupation}} = \{\text{salesman}, \text{engineer}\}$,$V_{\text{domicile}} = \{\text{Osaka}, \text{Tokyo},\text{Fukuoka}\}$ and $V_{\text{evaluation}} = \{\text{high}, \text{low}, \text{medium}\}$. The information function f is defined by the table as $f(o_1, \text{occupation}) = \text{salesman}$, $f(o_3, \text{domicile}) = \text{Tokyo}$, $f(o_4, \text{evaluation}) = \text{medium}$, and so on. Using decision attribute value $v_j^d \in V_d$, we define

Table 1 An example of decision table

Object	Sex	Occupation	Domicile	Evaluation
o_1	male	salesman	Osaka	high
o_2	male	salesman	Tokyo	high
o_3	female	salesman	Tokyo	low
o_4	female	engineer	Fukuoka	medium
o_5	male	engineer	Osaka	low
o_6	male	salesman	Fukuoka	high

decision class $D_j \subseteq U$ by $D_j = \{u \in U \mid f(u,d) = v_j^d\}$, $j = 1, 2, \dots, p$. On the other hand, equivalence classes are defined by using condition attributes $A \subseteq C$ as $[u]_A = \{x \in U \mid f(x,a) = f(u,a),\ \forall a \in A\}$. $[u]_A$ is the set of objects in U having the same attribute values as u in A.

Under a decision table, given a set of condition attributes $A \subseteq C$, the lower and upper approximations of an object set $X \subseteq U$ is defined by

$$A_*(X) = \{u \in U \mid [u]_A \subseteq X\}, \quad A^*(X) = \{u \in U \mid [u]_A \cap X \neq \emptyset\}. \tag{1}$$

$[u]_A \subseteq X$ implies that all objects in U having the same attribute values as u in A belong to X. On the other hand, $[u]_A \cap X \neq \emptyset$ implies that there exists at least one object in X which has the same attribute values as u in A. Thus, $A_*(X)$ is composed of consistent members of X and $A^*(X)$ is composed of possible members of X. The pair $(A_*(X), A^*(X))$ is called the rough set of X with respect to $A \subseteq C$. In the decision table shown in Table 1, for example, we obtain $A_*(D_1) = \{o_1.o_6\}$ and $A^*(D_1) = \{o_1, o_2, o_3, o_6\}$, where we define $A = \{\text{occupation}, \text{domicile}\}$ and $D_1 = \{o \in U \mid f(o,d) = \text{high}\} = \{o_1, o_2, o_6\}$.

In rough set approaches (Pawlak 1991), the attribute reduction and the minimal length rule induction are investigated well. In this paper, we utilize a rule induction method based on rough sets. In the decision table shown in Table 1, we find a decision rule “if sex = male, occupation = salesman and domicile=Osaka then evaluation = high”. However, this is not a minimal because we find a decision rule “if sex = male and occupation = salesman then evaluation = high” which has a shorter premise. This decision rule is a minimal length rule because there is no rule with shorter premise. In the rough set approach, such minimal length rules are induced. We use MLEM2 algorithm (Grzymala-Busse 2002) for inducing minimal length rules from a given decision table. By this algorithm, we obtain minimal set of rules with minimal conditions which can explain all objects in lower approximations of X under a given decision table. MLEM2 is a rule induction algorithm accommodating numerical/ordinal attributes as well as nominal/categorical attributes. Therefore, rules of the form of “if $v_1^{\mathrm{L}} \leq f(u,a_1) \leq v_1^{\mathrm{R}}, \cdots$ and $v_s^{\mathrm{L}} \leq f(u,a_s) \leq v_s^{\mathrm{R}}$ and $f(u,b_1) = w_1 \cdots f(u,b_t) = w_t$ then $u \in X$”, where $a_1, \dots, a_s \in C$ are numerical/ordinal attributes and $b_1, \dots, b_t \in C$ are nominal/categorical attributes. Namely, MLEM2 treats numerical/ordinal condition attributes by putting the condition expressed by intervals of attribute values in the premises of rules. For X, we usually substitute a decision class D_j, $j \in \{1, 2, \dots, p\}$. Then we apply MLEM2 usually to each decision class D_j, $j = 1, 2, \dots, p$ so that we obtain a minimal set of rules with minimal conditions, ‘if condition H_j^l is satisfied then the object is in a class D_j’, $l = 1, 2, \dots, m_j$ $j = 1, 2, \dots, p$, where H_j^l is a minimal condition described by condition attributes to infer the membership of D_j and m_j is the number of rules inferring the membership of D_j in the minimal set of rules induced by MLEM2. Using the set of those induced decision rules, we can build a classifier based on LERS system (Grzymala-Busse 1992).

When $p > 2$ (p is the number of decision classes), we may induce rules for a union of D_j’s by MLEM2 (see (Hamakawa and Inuiguchi 2014)). For example, when we consider a union $D_1 \cup D_2$, we can induce rules ‘if condition H_{12}^l is satisfied then the object belongs to union $D_1 \cup D_2$’, $l = 1, 2, \dots, m_{12}$, where m_{12} is the number of rules induced by MLEM2 with respect to $D_1 \cup D_2$. We may consider any union of $\bigcup_{j \in J} D_j$ and we obtain rules inferring the membership of the union, i.e., ‘if condition H_J^l is satisfied then the object belongs to union $\bigcup_{j \in J} D_j$’, $l = 1, 2, \dots, m_J$, where $j \in J \subseteq \{1, 2, \dots, p\}$ and m_J is the number of rules induced by MLEM2 with respect to $\bigcup_{j \in J} D_j$. If the condition H_J^l of such a rule is satisfied with an object, from the rule, we know that the object belongs to one of decision classes D_j, $j \in J$. Namely, from the rule, we know imprecisely the class including the object.

The decision rule having a union of decision classes in its conclusion is called an ‘imprecise decision rule’, or shortly, an ‘imprecise rule’. In contrast with an imprecise rule, we call usual decision rules with a single decision class ‘precise decision rules’, or shortly, ‘precise rules’. Although each imprecise rule can conclude the decision class only imprecisely, it works well for object classification by intersecting the imprecise conclusions of different imprecise rules. For example, the imprecise rules shown in Table 2 can be obtained from Table 1. We note that in Table 2 the notation such as ‘evaluation = (low or medium)’ is used instead of ‘the object belongs to the union of class [low] and class [medium]’ for the sake of simplicity. Under imprecise rules in Table 2, we can conclude “an object with occupation = engineer and domicile = Fukuoka is evaluated as medium”, “an object with occupation = salesman and domicile = Fukuoka is evaluated as high”, and so on. Therefore, we may build a classifier using imprecise rules in a similar way to LERS [see (Hamakawa and Inuiguchi 2014)], where LERS provides a method for classifying any object by using

Table 2 Imprecise rules found in Table 1

Name	Imprecise rule
I1	If sex = female then evaluation = (low or medium)
I2	If occupation = engineer then evaluation = (low or medium)
I3	If sex = male then evaluation = (low or high)
I4	If occupation = salesman then evaluation = (low or high)
I5	If domicile = Fukuoka, then evaluation = (medium or high)
I6	If sex = male and occupation = salesman then evaluation=(medium or high)

precise rules. Hamakawa and Inuiguchi (2014) demonstrated that the classifier using imprecise rules performs better than the classifier using precise rules.

Consider a situation where the publication of induced decision rules are requested. Such rule publication may be important and necessary for showing the fair and reasonable treatment of objects as well as for knowledge exchange. If rule r is supported only by a few objects, attribute values shown in rule r identify a few objects. If sensitive data are included in rule r, this identification invades privacy. From the viewpoint of data privacy, we cannot publish such rules, i.e., rules supported by a few objects. On the other hand, hiding such rules due to the privacy protection may bring an insufficient knowledge exchange and a sense of distrust. Because of missing rules, the classification of some objects cannot be explained well by the published rules. We call rule r a k-anonymous rule if r is supported by not less than k objects, where k is the minimally required number to protect the privacy.

As the number of objects in a union of classes is bigger than the number of objects in a single class, objects supporting an imprecise rule r inferring the membership of a union $\bigcup_{j\in J} D_j$ ($J \subseteq \{1,2,\dots,p\}$) is usually more than a precise rule r' inferring the membership of D_i ($i \in J$). As J enlarges, $\bigcup_{j\in J} D_j$ become large and thus the number of objects supporting imprecise rules inferring the membership of $\bigcup_{j\in J} D_j$ increases. Therefore, imprecise rules satisfies the k-anonymity more often than precise rules. If the objects supporting precise rules violating the k-anonymity are classified correctly by some k-anonymous imprecise rules, the replacement of precise rules violating the k-anonymity with those k-anonymous imprecise rules in the published rules can keep the classification quality. From this point of view, Inuiguchi et al. (2015) proposed an induction method for k-anonymous rules by utilizing imprecise rules, and showed the advantage of this approach.

3 The proposed data anonymization methods

We propose a data anonymization method for decision tables based on k-anonymous rules as an extension of the method proposed by Inuiguchi and Washimi (2019). We assume that the original decision table has more than two decision classes ($p > 2$) and that a set of k-anonymous rules induced from it is given (if not, we may induce those rules by the k-anonymous rule induction method (Inuiguchi et al. 2015)). The proposed anonymized table is composed of patterns each of which is expressed by a combination of imprecise attribute values. It is produced by embedding k-anonymous rules. Each imprecise pattern in the proposed table has at least k supporting objects in the original decision table (we call this property "k-commonality"). In this way, the proposed anonymized table preserves data privacy and is called a 'k-common pattern table'.

The k-common pattern table is similar to a set of k-anonymous rules but they are different. The k-common pattern table is composed of patterns. A pattern in the k-common pattern table specifies all condition and decision attribute values by value sets. On the other hand, a k-anonymous rule usually specify value sets of a part of condition attributes and the decision attribute. As described later, each pattern in the k-common pattern table is produced from a k-anonymous rule. However, there is no guarantee that all k-anonymous rules are used for producing patterns in the k-common pattern table. Moreover, the specified condition attribute value sets in k-anonymous rules can be a proper superset of the corresponding value sets of the pattern in the k-common pattern table. Usually only a part of k-anonymous rules are used to produce a k-common pattern table.

Before describing the algorithm for producing a k-common pattern table, we briefly describe the idea. Roughly speaking, a k-common pattern table is produced by embedding many of k-anonymous rules. Because k-anonymous rules necessary for the classification of objects consistent with the original table are embedded, it may preserve the quality of rules inducible from the proposed anonymized table. First, from a set of k-anonymous rules, a set of minimally necessary rules are selected for each object in the original table so as to specify its decision attribute value precisely. From each selected rule, we produce a pattern by putting the attribute values if they are specified in the rule, and a set of attribute values of k objects supporting the rule otherwise. We repeat this procedure for all selected rules of an object and for all objects in the original table, and as a result, we obtain a k-common pattern table. In this procedure, there are some options with different strategies. One is quality/usefulness-oriented strategy and the other is privacy protection-oriented strategy.

Let $Cl(u|R)$ be the set of the inferred values for decision attribute value of $u \in U$ under rule set R. The proposed procedure for building k-common pattern tables from a given decision table $DT = \langle U, C \cup \{d\}, V, f\rangle$ under a set $\mathcal{R}$ of k-anonymous rules is as follows:

(i) Let $cT = \langle\langle \mathcal{P}, C \cup \{d\}, V, \rho\rangle\rangle$ be a k-common pattern table corresponding to DT, where $\mathcal{P}$ is a set of patterns and $\rho : \mathcal{P} \times C \cup \{d\} \to \bigcup_{a\in C\cup\{d\}} 2^{V_a}$, 2^{V_a} is the power set of V_a, $a \in C \cup \{d\}$. Initialize $\mathcal{P} = \emptyset$.

(ii) For each object $u \in U$, obtain a minimal set $R(u) \subseteq \mathcal{R}$ of k-anonymous rules such that $Cl(u|R(u)) = Cl(u|\mathcal{R})$. Execute (a) and (b):

(a) If $R(u) = \emptyset$, terminate this procedure for $u \in U$.

(b) For each rule $r \in R(u)$, we select k objects supporting r and produce an imprecise pattern pt respectively by routines (s1)–(s3) and (s4)–(s7) described in what follows. Update $\mathcal{P} = \mathcal{P} \cup \{pt\}$. Return to (a).

In (ii), if the decision class of object u is estimated well by rule set $\mathcal{R}$, $Cl(u|\mathcal{R})$ is a singleton. Otherwise, $Cl(u|\mathcal{R})$ is a set of multiple decision attribute values or an emptyset. $Cl(u|\mathcal{R}) = \emptyset$ implies no matching rules in $\mathcal{R}$ for object u.

The number of elements of $\mathcal{P}$ can be larger than the number of elements in U. Namely, we obtain a larger table cT than the original decision table DT. The existence of k objects supporting a rule $r \in R(u)$ is guaranteed by the k-anonymity of r when $R(u) \neq \emptyset$.

In this paper, the following routine (s1)–(s3) is applied to the selection of k objects supporting r at (b) of (ii) in the proposed procedure:

(s1) Initialize $OB \subseteq U$ by the set of objects supporting r except u, and $O(r) = \emptyset$.

(s2) Select an object u' from OB which maximizes/minimizes the number of condition attribute values common in u and u'. If a tie occurs, select the first one among them. Update $O(r) = O(r) \cup \{u'\}$.

(s3) If $|O(r)| < k$, update $OB = OB - \{u'\}$ and return to (s2), where $|Y|$ is the cardinality of set Y.

At step (s2), there are two options in the selection of u'. If we select it with the maximal number of common condition attribute values, the obtained k-common pattern table preserves attribute values more precisely. On the other hand, if we select it with the minimal number of common condition attribute values, the obtained k-common pattern table becomes more ambiguous. Routine (s1)–(s3) is expressed as a flowchart in Fig. 1. Repeating routine (s1)–(s3) for each $r \in R(u)$, we obtain $O(r)$, $r \in R(u)$, where $O(r)$ is a set of k objects supporting rule r. Given $R(u)$ and $O(r)$, $r \in R(u)$, a pattern pt at (b) of (ii) in the proposed procedure is composed by routine (s4)–(s7) described later. By the following routine (s4)–(s7), a value set $\rho(pt, a)$ for each attribute $a \in C \cup \{d\}$ in pattern pt corresponding to $r \in R(u)$ is determined under given $R(u)$ and $O(r')$, $r' \in R(u)$.

(s4) Let $\tilde{C} = C$. The value set $\rho(pt, d)$ of decision attribute d in pt is defined by the set of decision attribute values specified in the conclusion of r.

(s5) Take a condition attribute $a \in \tilde{C}$.

(s5-a) If a appears in r, determine the value set $\rho(pt, a)$ by the value set specified in r for $a \in C$ and go to (s6).

(s5-b) If a is absent in all $r' \in R(u)$, determine the value set $\rho(pt, a)$ by the following way and go to (s6): if a is numerical/ordinal, $\rho(pt, a) = \bigcup_{r' \in R(u)} \left[\{f(u, a) \mid u \in O(r')\}\right]$, and if a is nominal, $\rho(pt, a) = \bigcup_{r' \in R(u)} \{f(u, a) \mid u \in O(r')\}$.

(s5-c) Determine the value set $\rho(pt, a)$ by the following way and go to (s6): if a is numerical/ordinal, $\rho(pt, a) = \left[\{f(u, a) \mid u \in O(r)\}\right]$, and if a is nominal, $\rho(pt, a) = \{f(u, a) \mid u \in O(r)\}$.

(s6) This step is optional. For each $r' \in R(u)$, update $\rho(pt, a)$ in the following way: if a appears in r' while not in that of r, update $\rho(pt, a)$ by a union of $\rho(pt, a)$ and the value set specified in r' for a.

(s7) Update $\tilde{C} = \tilde{C} - \{a\}$. If $\tilde{C} = \emptyset$, terminate this routine. Otherwise, return to (s5).

In routine (s4)–(s7), $\left[\{f(u, a) \mid u \in O(r)\}\right]$ for numerical/ordinal attribute a is the minimal interval covering $\{f(u, a) \mid u \in O(r)\}$.

Step (s6) is optional, if we adopt (s6), the k-common pattern table obtained has more imprecise values, i.e., the value set $\rho(pt, a)$ is larger. Thus it would protect the privacy more while deteriorate the quality/usefulness of the k-common pattern table. We note that some k-common patterns may

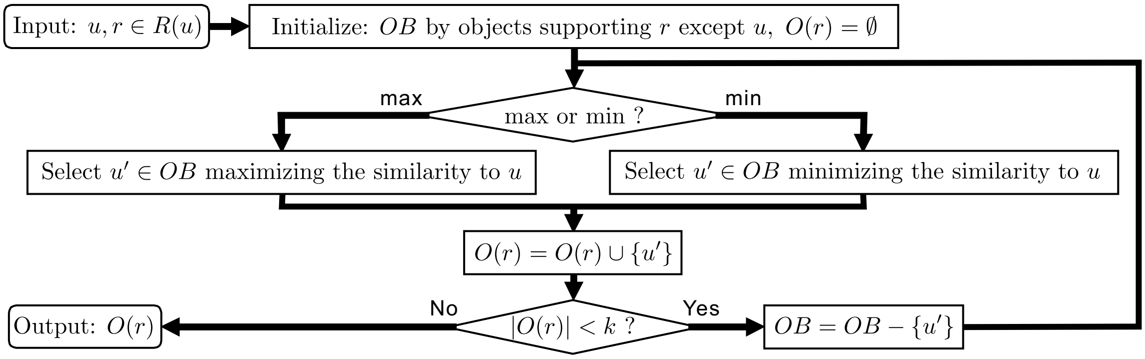
Input: $u, r \in R(u)$
Initialize: OB by objects supporting r except u, $O(r) = \emptyset$
max
max or min ?
min
Select $u' \in OB$ maximizing the similarity to u
Select $u' \in OB$ minimizing the similarity to u
$O(r) = O(r) \cup \{u'\}$
No
$|O(r)| < k$?
Yes
Output: $O(r)$
$OB = OB - \{u'\}$

Fig. 1 Determination of $O(r)$ under given u and $r \in R(u)$

be produced multiple times in the procedure. We may delete overlapped patterns. Routine (s4)–(s7) is expressed as a flowchart in Fig. 2.

As described in the previous section, we have options at steps (s2) and (s6) and thus we have four methods shown in Table 3. Namely, we consider four methods denoted by cT1, cT2, cT3 and cT4. Among those, cT1 corresponds to the one proposed by Inuiguchi and Washimi (Inuiguchi and Washimi 2019).

All imprecise rules shown in Table 2 are 2-anonymous rules. To illustrate the proposed approach, let us apply the procedure described above to the decision table shown in Table 1 using 2-anonymous rules shown in Table 2. Selecting rules and objects as shown in Table 4, we obtain 2-common pattern tables shown in Table 5. In the case of Table 1, option (s6) does not influence the resulting 2-common pattern tables. The underlined attribute values in Table 5 are specified by the used imprecise rules. As shown in Table 5, the obtained 2-common pattern tables cT3 and cT4 are more imprecise than cT1 and cT2. We note that the underlined values do not appear in the published anonymized table. We show them for easy verification of the resulting 2-common pattern tables.

Table 3 Four proposed methods

(s2)	Without (s6)	With (s6)
Maximize	cT1	cT2
Minimize	cT3	cT4

From both tables shown in Table 5, we may induce imprecise rules shown in Table 6, where we note that a condition on an attribute is satisfied if the attribute value set in the condition includes the attribute value set of a pattern. I8’ is induced only from cT1 and cT2 while I8” is induced only from cT3 and cT4. Rules I1’ to I6’ and I8’ are induced from Table 1 while rules I7’ to I8” are more imprecise than those induced from Table 1. From pattern pt'_{10}, no rule is induced. We observe that rules are preserved in the 2-common pattern tables although there are less rules and weaker. Now let us check the privacy protection of 2-common pattern tables in Table 5. Because the original table in Table 1 is very small, the privacy protection of this example is not very remarkable. However, we can see that the 2-common pattern tables protect some data. For example, if we know sex = female and occupation = salesman, we infer domicile = (Tokyo or Fukuoka) and evaluation = (low or medium) in Table 5 while we obtain domicile = Tokyo and evaluation = low in Table 1. Similarly, if we know sex = male and occupation = engineer, we infer domicile = (Osaka or Fukuoka) and evaluation = (low or medium) in Table 5 while we obtain domicile = Osaka and evaluation = low in Table 1. As demonstrated above, 2-common pattern tables preserve

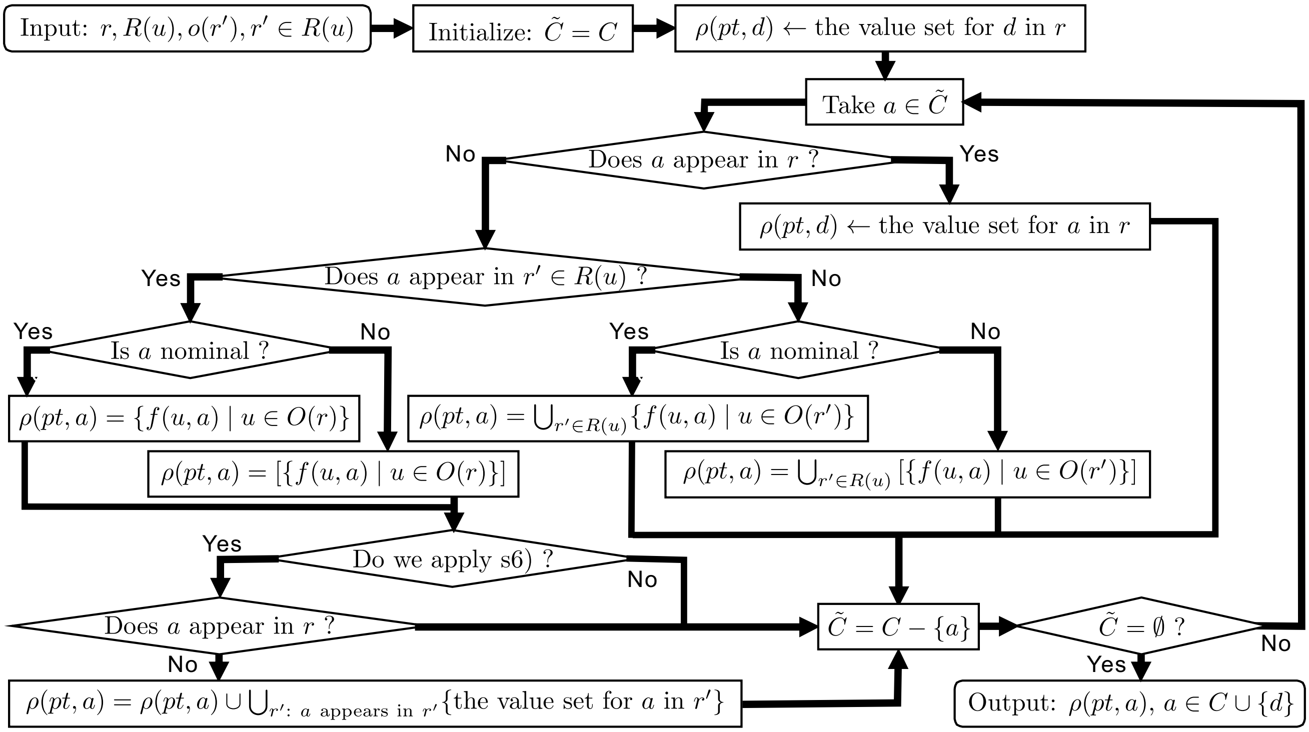
Input: $r, R(u), o(r'), r' \in R(u)$
Initialize: $\tilde{C} = C$
$\rho(pt, d) \leftarrow$ the value set for d in r
Take $a \in \tilde{C}$
No
Does a appear in r ?
Yes
$\rho(pt, d) \leftarrow$ the value set for a in r
Yes
Does a appear in $r' \in R(u)$?
No
Yes
Is a nominal ?
No
Yes
Is a nominal ?
No
$\rho(pt, a) = \{f(u, a) \mid u \in O(r)\}$
$\rho(pt, a) = \bigcup_{r' \in R(u)} \{f(u, a) \mid u \in O(r')\}$
$\rho(pt, a) = [\{f(u, a) \mid u \in O(r)\}]$
$\rho(pt, a) = \bigcup_{r' \in R(u)} [\{f(u, a) \mid u \in O(r')\}]$
Yes
Do we apply s6) ?
No
Does a appear in r ?
No
$\tilde{C} = C - \{a\}$
$\tilde{C} = \emptyset$?
No
Yes
$\rho(pt, a) = \rho(pt, a) \cup \bigcup_{r':\ a \text{ appears in } r'} \{$the value set for a in $r'\}$
Output: $\rho(pt, a),\ a \in C \cup \{d\}$

Fig. 2 Determination of pt under given $R(u)$ and $O(r')$, $r \in R(u)$

Table 4 Selected rules and objects

cT1 and cT2				cT3 and cT4			
Object u	Rule	Object u'	Pattern	Object u	Rule	Object u'	Pattern
o_1	I3	o_2	pt_1	o_1	I3	o_2	pt'_1
	I6	o_6	pt_2		I6	o_6	pt'_2
o_2	I4	o_3	pt_3	o_2	I4	o_3	pt'_3
	I6	o_6	pt_4		I6	o_6	pt'_4
o_3	I1	o_4	pt_5	o_3	I1	o_4	pt'_5
	I4	o_2	pt_3		I4	o_1	pt'_6
o_4	I2	o_5	pt_6	o_4	I2	o_5	pt'_7
	I5	o_6	pt_7		I5	o_6	pt'_8
o_5	I2	o_4	pt_6	o_5	I2	o_4	pt'_7
	I3	o_1	pt_8		I3	o_2	pt'_9
o_6	I3	o_1	pt_9	o_6	I3	o_5	pt'_{10}
	I5	o_4	pt_7		I5	o_4	pt'_8

Table 5 2-Common pattern tables obtained from Table 1

Pattern	Sex	Occupation	Domicile	Evaluation
cT1 and cT2				
pt_1	male	salesman	Tokyo or Osaka	low or high
pt_2	male	salesman	Osaka or Fukuoka	medium or high
pt_3	male or female	salesman	Tokyo	low or high
pt_4	male	salesman	Tokyo or Fukuoka	medium or high
pt_5	female	salesman or engineer	Tokyo or Fukuoka	low or medium
pt_6	male or female	engineer	Osaka or Fukuoka	low or medium
pt_7	male or female	salesman or engineer	Fukuoka	medium or high
pt_8	male	salesman or engineer	Osaka	low or high
pt_9	male	salesman	Osaka or Fukuoka	low or high
cT3 and cT4				
pt'_1	male	salesman	Tokyo or Osaka	low or high
pt'_2	male	salesman	Osaka or Fukuoka	medium or high
pt'_3	male or female	salesman	Tokyo	low or high
pt'_4	male	salesman	Tokyo or Fukuoka	medium or high
pt'_5	female	salesman or engineer	Tokyo or Fukuoka	low or medium
pt'_6	male or female	salesman	Tokyo or Osaka	low or high
pt'_7	male or female	engineer	Osaka or Fukuoka	low or medium
pt'_8	male or female	salesman or engineer	Fukuoka	medium or high
pt'_9	male	salesman or engineer	Tokyo or Osaka	low or high
pt'_{10}	male	salesman or engineer	Osaka or Fukuoka	low or high

rules inducible from the original table to a certain extent and preserve privacy more than the original table.

4 Numerical experiments

4.1 Outline

We investigate the performances of the proposed anonymized tables and compare them with the conventional anonymized tables. As the conventional anonymized tables, we use tables obtained by Mondrian (LeFevre et al. 2005; Torra 2017), a method achieving k-anonymity for multidimensional records described in terms of several condition attributes. Mondrian is a greedy partitioning algorithm recursively selects a condition attribute to partition into two sets with the same size until no further partition is needed (or possible). Mondrian is usually applied to all objects in the given decision table. However, it easily destroys characteristic patterns of decision classes described by condition attributes. Therefore, we consider also the application

Table 6 Imprecise rules found in Table 5

Name	Imprecise rule
I1′	If sex = female then evaluation = (low or medium)
I2′	If occupation = engineer then evaluation = (low or medium)
I3′	If sex = male and domicile = (Tokyo or Osaka) then evaluation = (low or high)
I4″	If domicile = Tokyo then evaluation = (low or high)
I5′	If domicile = Fukuoka, then evaluation = (medium or high)
I6′	If occupation = salesman and domicile = (Tokyo or Osaka) then evaluation = (low or high)
I7′	If sex = male and domicile = (Tokyo or Fukuoka) then evaluation = (medium or high)
I8′	If occupation = salesman and domicile = (Osaka or Fukuoka) then evaluation = high (from cT1 and cT2)
I8″	If occupation = salesman and domicile = (Osaka or Fukuoka) then evaluation = (medium or high) (from cT3 and cT4)

Table 7 Datasets

Dataset	$\|U\|$	$\|C\|$	$\|V_d\|$	Attribute
Car	1728	6	4	Ordinal
Hayes-roth	160	5	3	Ordinal
Iris	150	4	3	Numerical
Zoo	101	16	7	Nominal

of Mondrian to each decision class in order to increase the preservation probability of the characteristic patterns, i.e., to preserve the quality. The usual application of Mondrian is denoted by M1 while the application of Mondrian to each decision class is denoted by M2. We use four datasets shown in Table 7 obtained from the UCI Machine Learning Repository (Dua and Graff 2019). We assume that all attribute values should be anonymized. We apply 10-fold cross validation methods 10 times for each dataset.

4.2 Quality/usefulness of anonymized table

The proposed k-common pattern tables have good property that they can estimate the correct decision attribute values of objects in the original decision table if the objects are expressed by k-anonymous rules. Namely, the correct values are known by intersecting the value sets of decision attribute of patterns to which objects are matched. The k-anonymous tables obtained by Mondrian do not always have this property. However, for new objects, the classification ability based on the proposed k-common pattern tables remains uncertain.

We suppose that the analysts who can access only anonymized tables are interested in knowing the relations between condition and decision attributes by inducing decision rules. Therefore, we evaluate the usefulness of the anonymized data tables by the quality of induced decision rules. To measure the quality of induced decision rules, we use the classification accuracy of a classifier composed of these induced decision rules. For the classifier, we utilize LERS (Grzymala-Busse 1992, 2002) which is a method for classifying any object by using induced decision rules. Moreover, we assume that decision rules are induced by the following procedure based on MLEM2 (Grzymala-Busse 2002) with the initial set of elementary conditions composed of (i) "$f(x,a) \in S$" if a is a nominal/categorical attribute and the set S of condition attribute values appears in the given table and (ii) "$f(x,a) \geq v^{\mathrm{L}}$" and "$f(x,a) \leq v^{\mathrm{R}}$" if a is a numerical/ordinal attribute and the interval $[v^{\mathrm{L}}, v^{\mathrm{R}}]$ appears in the given table:

(r1) Set $j := 1$. Induce set R_1 of precise rules for each decision class D_i, $i \in \{1,2,\dots,p\}$ from a set of all patterns/objects Q_j.

(r2) Erase patterns/objects explained by rules in R_j. Let Q_{j+1} be the remaining patterns. If $Q_{j+1} = \emptyset$ or $j \geq n$, terminate this procedure.

(r3) Update $j := j + 1$. Induce set R_j of imprecise rules for all combinations of j decision classes D_i, $i \in \{1,2,\dots,p\}$ from a set of patterns/objects Q_j. Return to (r2).

The obtained results are shown in Table 8. In Table 8, 'Ori' stands for the original decision table. The values shown in rows of 'Ori' and 'k-Ori' are the classification accuracies of classifiers composed of decision rules induced by MLEM2 (Grzymala-Busse 2002) and those of classifiers composed of k-anonymous rules induced from the original decision table, respectively. We note that, as observed in reference (Inuiguchi et al. 2015), 'k-Ori' is sometimes better than 'Ori' because of the following reasons: (I) the number of k-anonymous rules can be more than that of decision rules induced by MLEM2 if k is small, and (II) a k-anonymous rule is supported by not less than k objects and thus usually more general than a decision rule induced by MLEM2. Indeed, we observe such results in datasets 'car' and 'hayes-roth' with $k = 5$ and 'car' with $k = 10$. However, as k increases, the number of k-anonymous rules decreases

Table 8 Classification accuracy of induced rules from tables

	$k = 5$				$k = 10$			
	Car	Hayes-roth	Iris	Zoo	Car	Hayes-roth	Iris	Zoo
Ori	0.9867	0.8131	0.9287	0.9643	0.9867	0.8131	0.9287	0.9643
k-Ori	0.9868	0.8400	0.9173	0.9612	0.9897	0.7106	0.9173	0.9437
cT1	0.8854	0.7994	0.9213	0.8583	0.8997	0.6369	0.9233	0.7239
cT2	0.8859	0.8000	0.9213	0.8650	0.8990	0.6394	0.9207	0.7369
cT3	0.8491	0.7969	0.9100	0.7397	0.8633	0.6200	0.9107	0.6739
cT4	0.8474	0.8000	0.9133	0.7023	0.8663	0.6275	0.9213	0.6656
M1	0.7011	0.1925	0.3413	0.0859	0.1152	0.1925	0.3447	0.0554
M2	0.7000	0.4531	0.8713	0.6780	0.7000	0.2806	0.7720	0.4065

so that the classification ability of a set of k-anonymous rules decreases.

As shown in Table 8, the rules induced from the proposed k-common pattern tables cT1, cT2, cT3 and cT4 are much better than those induced from k-anonymized tables M1 and M2 based on Mondrian. Tables cT3 and cT4 are less useful than tables cT1 and cT2 because the classification accuracies of the induced rules are worse. The differences between tables cT1 and cT2 as well as between tables cT3 and cT4 are not very big and we cannot say which tables are better. This implies that the adoption of maximization or minimization at step (s2) influences the usefulness of the obtained anonymized table and that the adoption of maximization is preferable. Moreover, although the classification accuracies of the rules induced from the proposed tables are worse than those induced from the original table, the classification accuracies do not degrade so much (except for the 'hayes-roth' data set with k = 10).

4.3 Privacy protection ability against a shallow attack

To evaluate the privacy protection ability, we investigate to what extent the value of a condition/decision attribute is correctly estimated from values of l condition attributes of an object, where the values of l condition attributes are supposed to be known by the attacker. We consider two estimation methods for the attribute values called shallow and deep attacks. A shallow attack tries to find unknown attribute values of a little known object without a big effort. On the other hand, a deep attack tries to find unknown attribute values of a little known object with a brute-force serach. A shallow attack is supposed to be used when the attacker is intersted in unknown attribute values only to some extent so that s/he does not invest a lot of her/his time and effort. A deep attack is supposed to be used when the attacker is interested in unknown attribute values very much so that s/he is willing to spend a lot of her/his time and effort. We note that the proposed k-common pattern tables cT1, cT2, cT3 and cT4 and the k-anonymized table M2 are designed for preserving rules estimating decision attribute values from condition attribute values. Therefore, it is difficult to protect decision attribute values as the number l of known condition attribute values increases.

In the shallow attack, we first select all patterns/objects that can take the revealed attribute values in the anonymized table. Then we collect the values of attribute we want to attack for those selected patterns/objects and take their intersection. If the intersection is a singleton and it is the correct attribute value of the object, the shallow attack is successful. For example, suppose that the domicile of a person listed in a common pattern table cT1 in Table 5 is revealed as 'Tokyo'. Then patterns pt_1, pt_3, pt_4 and pt_5 are selected because they can take 'Tokyo' for their domicile. Attacking on occupation, we obtain 'salesman' and 'salesman or engineer' from those patterns. Taking the intersection, we obtain 'salesman'. As a result, we successfully reveal that the person living in 'Tokyo' is 'salesman' which is correct as shown in Table 1. On the other hand, attacking on sex, we obtain 'male', 'male or female' and 'female' and the intersection is empty. Therefore, the attack on sex of that person is failed.

Because $|C|$ of the datasets treated in our experiments are not very large, we set $l = 2$. All possible attacks are evaluated. The obtained results are shown in Tables 9 and 10. Table 9 shows the average ratios of successful attacks on condition attributes while Table 10 shows the average ratios of successful attacks on decision attributes. In those tables, k stands for the parameters of k-commonality as well as k-anonymity.

As shown in Table 9, the k-anonymized table M1 protects the condition attribute values more than the proposed k-common pattern tables cT1, cT2, cT3 and cT4 except for Dataset 'hayes-roth'. The privacy protection ability of the k-anonymized table M2 is comparable with that of the k-common pattern tables cT3 and cT4. The proposed k-common pattern tables cT1, cT2, cT3 and cT4 perform much better in Dataset 'hayes-roth' than the k-anonymized tables M1 and M2. Roughly speaking, the privacy protection abilities of the proposed k-common pattern tables are not very bad although they are worse than the k-anonymized table M1.

Table 9 Success ratios of attacking condition attribute values by the shallow attack

k	cT1	cT2	cT3	cT4	M1	M2
Dataset: car						
2	0.1806	0.1815	0.1842	0.1848	0.0418	0.0504
3	0.1736	0.1721	0.1770	0.1790	0.0419	0.1438
4	0.1541	0.1540	0.1539	0.1549	0.0845	0.1907
5	0.1395	0.1395	0.1408	0.1432	0.0845	0.2487
10	0.0598	0.0586	0.0639	0.0639	0.2066	0.2262
15	0.0519	0.0510	0.0570	0.0569	0.0265	0.2023
Dataset: hayes-roth						
2	0.0412	0.0409	0.0549	0.0532	0.0299	0.0986
3	0.0394	0.0402	0.0570	0.0563	0.2018	0.2579
4	0.0581	0.0540	0.0752	0.0723	0.2018	0.4326
5	0.0760	0.0744	0.0812	0.0813	0.5223	0.4228
10	0.1431	0.1345	0.1024	0.1066	0.4113	0.0453
15	0.3017	0.3018	0.3393	0.3392	0.4113	0.0196
Dataset: iris						
2	0.3053	0.2870	0.2447	0.2345	0.2947	0.1767
3	0.3539	0.3071	0.1622	0.1598	0.0032	0.0627
4	0.3245	0.2915	0.0089	0.0097	0.0032	0.0000
5	0.3036	0.2684	0.0073	0.0065	0.0000	0.0000
10	0.0564	0.0508	0.0050	0.0050	0.0000	0.0000
15	0.0128	0.0128	0.0000	0.0000	0.0000	0.0000
Dataset: zoo						
2	0.7010	0.7001	0.6902	0.6890	0.7028	0.7986
3	0.7249	0.7158	0.7147	0.7093	0.6973	0.7356
4	0.7421	0.7324	0.7079	0.7056	0.5413	0.6810
5	0.7552	0.7332	0.7060	0.6936	0.5413	0.5565
10	0.7588	0.7280	0.6965	0.6857	0.2394	0.3297
15	0.7417	0.7031	0.6834	0.6656	0.0673	0.2061

Now we see the results shown in Table 10. The *k*-anonymized table M1 protects the condition attribute values more than the proposed *k*-common pattern tables cT1, cT2, cT3 and cT4 in all datasets, because table M1 is anonymized without the consideration of the decision classes, i.e., decision attribute values. Therefore, from the viewpoint of privacy protection, table M1 is the best but its usefulness in rule induction is the worst as we have seen in the previous subsection. Comparing the results to those of *k*-anonymized table M2, the privacy protection abilities of the proposed *k*-common pattern tables cT1, cT2, cT3 and cT4 are comparable. The proposed tables cT1, cT2, cT3 and cT4 are better in datasets 'car' and 'hayes-roth'. Among the proposed *k*-common pattern tables cT1, cT2, cT3 and cT4, tables cT3 and cT4 are better than tables cT1 and cT2. Between cT1 and cT2, as well as between cT3 and cT4, cT2 is better than cT1 and cT4 is better than cT3. As a summary, it is preferable from a privacy perspective to choose 'minimize' at step (s2) and adopting step (s6) in the proposed algorithm to build a *k*-common pattern table.

4.4 Privacy protection ability against a deep attack

In this subsection, we describe the results of the deep attack. In the shallow attack, we consider the intersection of decision attribute value sets of all patterns/objects match to the known condition attribute values. However, we did not check the compatibility among selected patterns/objects. Namely, some of the selected patterns/objects can be incompatible, i.e., the intersection of value sets of some condition attribute can be empty while all of them have the known condition attribute values. In the deep attack, we take into account the compatibility among selected patterns/objects. The deep attack is launched only when the shallow attack fails. The procedure of the deep attack is as follows:

(d0) We assume that attribute $a \in C \cup \{d\}$ is attacked. Let U_a be the set of patterns/objects that match to the known attribute values.

(d1) Calculate a minimal set $P \subseteq U_a$ of mutually compatible patterns/objects which uniquely specify a value $\bar{v}_a$ of a by their intersection.

Table 10 Success ratios of attacking decision attribute values by the shallow attack

k	cT1	cT2	cT3	cT4	M1	M2
Dataset: car						
2	0.2064	0.2064	0.2064	0.2064	0.0000	0.1913
3	0.2067	0.2067	0.2068	0.2068	0.0000	0.1913
4	0.2069	0.2068	0.2068	0.2068	0.0003	0.1914
5	0.2069	0.2069	0.2068	0.2068	0.0003	0.2220
10	0.2070	0.2070	0.2070	0.2071	0.0033	0.3722
15	0.2074	0.2074	0.2072	0.2073	0.0000	0.4198
Dataset: hayes-roth						
2	0.3787	0.3783	0.3650	0.3647	0.0007	0.3563
3	0.3744	0.3628	0.3632	0.3531	0.0004	0.3451
4	0.3683	0.3572	0.3597	0.3514	0.0004	0.3473
5	0.3647	0.3569	0.3595	0.3544	0.0000	0.3854
10	0.3613	0.3584	0.3604	0.3619	0.0000	0.4317
15	0.3142	0.3142	0.3042	0.3042	0.0000	0.4057
Dataset: iris						
2	0.6061	0.6067	0.5967	0.5967	0.0175	0.7662
3	0.6682	0.6654	0.6657	0.6682	0.0011	0.7506
4	0.6848	0.6816	0.6889	0.6879	0.0011	0.7334
5	0.6937	0.6947	0.6969	0.6954	0.0000	0.7334
10	0.7345	0.7349	0.7328	0.7348	0.0000	0.7121
15	0.7433	0.7470	0.7445	0.7445	0.0000	0.6981
Dataset: zoo						
2	0.2890	0.2868	0.1958	0.1959	0.0012	0.2581
3	0.2784	0.2282	0.1847	0.1716	0.0001	0.2453
4	0.2586	0.1785	0.1628	0.1373	0.0000	0.2507
5	0.2415	0.1184	0.1638	0.1243	0.0000	0.2868
10	0.2211	0.1090	0.1539	0.1465	0.0000	0.4314
15	0.1889	0.1169	0.1577	0.1503	0.0000	0.5747

(d2) Obtain a set $Q \subseteq U_a$ by collecting patterns/objects whose decision attribute value sets do not include $\bar{v}_1$.

(d3) For each pattern/object in Q, we check whether the values of condition attributes except attribute a match to all patterns/objects in P or not. If there is such a pattern/object in Q, the value of attribute a cannot be uniquely determined and thus, the deep attack fails.

(d4) Calculate a minimal set $S \subseteq Q$ of mutually compatible patterns/objects which uniquely specify a value $\hat{v}_a$ of a by their intersection. If there is such a set S, the deep attack fails because there are at least two possibilities of the value of a, i.e., $\bar{v}_a$ and $\hat{v}_a$.

(d5) If the specified value $\bar{v}_a$ of attribute a is correct, the deep attack is successful.

The set of patterns/objects P and S at step (d1) and (d4) can be obtained, for example, by a branch and bound method.

Because the deep attack requires a very big computational effort, we apply the deep attack only to a small-sized decision table composed of 15 objects randomly sampled from the original dataset. Moreover, we examine the protection abilities of anonymized tables with $k = 2$, 3 and 4 against the deep attack only for the three data sets, 'car', 'hayes-roth' and 'iris' which have small numbers of attribute values both for condition and decision attributes. The deep attack is applied when the shallow attack fails.

The results of the deep attack are shown in Tables 11 and 12. Table 11 shows the obtained success ratios of attacking condition attribute values while Table 12 shows the obtained success ratios of attacking decision attribute values. In both tables, we show the results of the shallow attack and the results of the deep attack so that we can see to what extent the deep attack works.

As shown in Tables 11 and 12, the deep attack reveals significantly more attribute values than the shallow attack. In Table 11, we observe that the proposed k-common pattern tables cT1, cT2, cT3 and cT4 generally protect the attribute values less than k-anonymous tables M1 and M2. However, the differences are not very big and the proposed k-common pattern tables are better in some cases, because the size of the original decision table is small. On the other hand, in

Table 11 Success ratios of attacking condition attribute values by shallow and deep attacks

k	cT1	cT2	cT3	cT4	M1	M2
Dataset: car (shallow attack)						
2	0.3002	0.2954	0.2895	0.2872	0.3272	0.3350
3	0.3042	0.2867	0.2816	0.2623	0.2343	0.2388
4	0.2940	0.2821	0.2531	0.2400	0.1828	0.1459
Dataset: car (deep attack)						
2	0.3450	0.3323	0.2921	0.2930	0.3372	0.3366
3	0.3100	0.2849	0.2833	0.2588	0.2467	0.2388
4	0.3060	0.2816	0.2640	0.2347	0.1940	0.1459
Dataset: hayes-roth (shallow attack)						
2	0.5078	0.4823	0.4557	0.4443	0.3608	0.3666
3	0.4368	0.4159	0.3991	0.3867	0.2661	0.2381
4	0.4209	0.4102	0.3749	0.3653	0.2033	0.1212
Dataset: hayes-roth (deep attack)						
2	0.6326	0.6000	0.5667	0.5375	0.3894	0.3787
3	0.4957	0.4625	0.4487	0.4293	0.2861	0.2397
4	0.4757	0.4704	0.4471	0.4461	0.3747	0.1697
Dataset: iris (shallow attack)						
2	0.0638	0.0639	0.0163	0.0163	0.0194	0.0108
3	0.0096	0.0096	0.0026	0.0026	0.0005	0.0002
4	0.0017	0.0016	0.0014	0.0014	0.0001	0.0001
Dataset: iris (deep attack)						
2	0.0723	0.0724	0.0182	0.0182	0.0210	0.0108
3	0.0150	0.0141	0.0027	0.0027	0.0005	0.0002
4	0.0029	0.0026	0.0015	0.0015	0.0001	0.0001

Table 12, we observe that the proposed *k*-common pattern tables cT1, cT2, cT3 and cT4 protect the attribute values more than *k*-anonymous table M2 but less than *k*-anonymous table M1. In Tables 12 and 11, we observe that the protection ability increases in the order of cT1, cT2, cT3 and cT4. This implies that the advantages of the adoptions of 'minimize' at step (s2) and step (s6) in the proposed data anonymization algorithm are more remarkable in those tables.

5 Concluding remarks

In this paper, we proposed an anonymization approach using imprecise rules (Inuiguchi et al. 2015; Hamakawa and Inuiguchi 2014). In the proposed approach, we first obtain *k*-anonymous rules each of which is supported by at least *k* objects in the given decision table. Then using the anonymized rules, the given decision table is anonymized by replacing anonymizable objects with *k*-common patterns and deleting non-anonymizable objects. Then the obtained table by a proposed method is called a *k*-common pattern table. Four *k*-commonization methods are proposed. They are different in the value specification level of condition attributes unspecified in the underlying *k*-anonymous rules. By means of numerical experiments, we demonstrated that the usefulness of the proposed *k*-common pattern tables in rule mining is much better than the previous *k*-anonymized decision table based on Mondrian. However, the privacy protection abilities of the proposed tables are worse than a *k*-anonymized decision table obtained by Mondrian without consideration of classification by the decision attribute. The approach using Mondrian destroys the classification possibility of the original table drastically. The decision attribute value protection abilities of the proposed tables are comparable with a *k*-anonymized decision table obtained by Mondrian with consideration of classification by the decision attribute. Nevertheless, their condition attribute value protection abilities are worse. Considering the high performance of the usefulness in rule induction, the proposed *k*-common pattern tables are applicable especially when the decision attribute values are sensitive and should be protected.

We may increase the privacy protection ability by introducing imprecise condition values for nominal/categorical condition attributes when we induce the set of *k*-anonymous rules. Moreover, the proposed approach is applicable only when we have more than two decision attribute values. One of the future topics is to extend the proposed approach to cases where the decision attribute values take only two values.

Table 12 Success ratios of attacking decision attribute values by shallow and deep attacks

k	cT1	cT2	cT3	cT4	M1	M2
Dataset: car (shallow attack)						
2	0.5449	0.5414	0.5444	0.5512	0.3052	0.7146
3	0.5524	0.5500	0.5736	0.5682	0.1751	0.7765
4	0.6127	0.6150	0.6642	0.6610	0.1373	0.8331
Dataset: car (deep attack)						
2	0.5728	0.5664	0.5431	0.5451	0.3052	0.7146
3	0.5762	0.5707	0.5663	0.5668	0.1751	0.7765
4	0.6455	0.6478	0.6710	0.6684	0.1884	0.8240
Dataset: hayes-roth (shallow attack)						
2	0.6113	0.6016	0.5800	0.5736	0.1108	0.5187
3	0.5322	0.5280	0.4928	0.4920	0.0250	0.5302
4	0.3918	0.3897	0.3331	0.3327	0.0249	0.5170
Dataset: hayes-roth (deep attack)						
2	0.6683	0.6467	0.6238	0.6120	0.1118	0.5187
3	0.5661	0.5494	0.5454	0.5342	0.0250	0.5302
4	0.3974	0.3900	0.3647	0.3679	0.0249	0.5170
Dataset: iris (shallow attack)						
2	0.8834	0.8833	0.8572	0.8572	0.0990	0.9371
3	0.8649	0.8643	0.8443	0.8435	0.0134	0.9259
4	0.8194	0.8181	0.8123	0.8115	0.0028	0.9286
Dataset: iris (deep attack)						
2	0.8430	0.8434	0.8421	0.8422	0.0990	0.9371
3	0.8386	0.8380	0.8409	0.8402	0.0134	0.9259
4	0.8189	0.8170	0.8221	0.8204	0.0028	0.9286

Acknowledgements This work is supported by JSPS KAKENHI Grant number JP18H01658.

References

Abidi B, Ben YS, Perera C (2019) Hybrid microaggregation for privacy preserving data mining. J Ambient Intell Human Comput. https://doi.org/10.1007/s12652-018-1122-7

Dua D, Graff C (2019) UCI machine learning repository. School of Information and Computer Science, University of California, Irvine, CA. http://archive.ics.uci.edu/ml

Evfimuievski A, Srikant R, Agrawal R, Gehrke J (2004) Privacy preserving mining of association rules. Inf Sci 29:343–364

Fung BCN, Wang K, Chen R, Yu PS (2010) Privacy-preserving data publishing: a survey of recent developments. ACM Comput Surv 42(4):Article 14

Grzymala-Busse JW (1992) LERS—a system for learning from examples based on rough sets. In: Słowinński R (ed) Intelligent decision support. Handbook of applications and advances of the rough sets theory. Kluwer, Dordrecht, pp 3–18

Grzymala-Busse JW (2003) MLEM2—discretization during rule induction. In: Klopotek MA, Wierzchon ST, Trojanowski K (eds) Intelligent information processing and web mining: Proceedings of the IIPWM 2003. Springer, Heidelberg, pp 499–508

Hamakawa T, Inuiguchi M (2014) On the utility of imprecise rules induced by MLEM2 in classification. In: 2014 IEEE international conference on granular computing (GrC 2014) Noboribetsu, pp 76–81

Inuiguchi M, Hamakawa T, Ubukata S (2015) Imprecise rules for data privacy. In: Ciucci D, Wang G, Mitra S, Wu W-Z (eds) Rough sets and knowledge technology: Proceedings of 10th international conference, LNCS 9436. Springer, Cham, pp 129–139

Inuiguchi M, Washimi K (2019) Utilization of imprecise rules for privacy protection. In: Seki H, Nguyen CH, Huynh V-N, Inuiguchi M (eds) Integrated uncertainty in knowledge modelling and decision making: Proceedings of 7th international symposium, LNCS 11471. Springer, Cham, pp 260–270

LeFevre K, DeWitt DJ, Ramarkrishnan R (2005) Multidimensional *k*-anonymity. Technical report 1521, University of Wisconsion, Wisconsin

Lekshmy PL, Rahiman MA (2019) A sanitization approach for privacy preserving data mining on social distributed environment. J Ambient Intell Human Comput. https://doi.org/10.1007/s12652-019-01335-w

Mendes R, Vilela JP (2017) Privacy-preserving data mining: methods, metrics, and applications. IEEE Access 5:10562–10582

Pawlak Z (1982) Rough sets. Int J Comput Inf Sci 11(5):341–356

Pawlak Z (1991) Rough sets: theoretical aspects of reasoning about data. Kluwer Academic Publishing, Dordrecht

Rizvi SJ, Haritsa JR (2002) Maintaining data privacy in association rule mining. In: VLDB '02 Proceedings of the 28th international conference on very large data bases, Hong Kong, pp 682–693

Torra V (2017) Data privacy: foundations, new developments and the big data challenge. Springer, Cham

Ye M, Wu X, Hu X, Hu D (2013) Anonymizing classification data using rough set theory. Knowl Based Syst 43:82–94

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.