Proportional representation to increase data utility in k-anonymous tables
Fabien Viton, Clémence Mauger, Gilles Dequen, Jean-Luc Guérin, Gaël Le Mahec

To cite this version:
Fabien Viton, Clémence Mauger, Gilles Dequen, Jean-Luc Guérin, Gaël Le Mahec. Proportional representation to increase data utility in k-anonymous tables. 26th IEEE Symposium on Computers and Communications, Sep 2021, Athènes, Greece. hal-03414033

HAL Id: hal-03414033
https://hal.archives-ouvertes.fr/hal-03414033
Submitted on 4 Nov 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Proportional representation to increase data utility in k-anonymous tables

Fabien Viton, Clémence Mauger, Gilles Dequen, Jean-Luc Guérin, Gaël Le Mahec

To cite this version:
Fabien Viton, Clémence Mauger, Gilles Dequen, Jean-Luc Guérin, Gaël Le Mahec. Proportional representation to increase data utility in k-anonymous tables. 26th IEEE Symposium on Computers and Communications, Sep 2021, Athènes, Greece. hal-03414033
Proportional representation to increase data utility in $k$-anonymous tables

Fabien Viton  
Laboratoire MIS  
Université de Picardie Jules Verne  
Amiens, France  
fabien.viton@u-picardie.fr

Clément Mauger  
Laboratoire MIS  
Université de Picardie Jules Verne  
Amiens, France  
clemence.mauger@u-picardie.fr

Gilles Dequen  
Laboratoire MIS  
Université de Picardie Jules Verne  
Amiens, France  
gilles.dequen@u-picardie.fr

Jean-Luc Guérin  
Laboratoire MIS  
Université de Picardie Jules Verne  
Amiens, France  
jean-luc.guerin@u-picardie.fr

Gaël Le Mahec  
Laboratoire MIS  
Université de Picardie Jules Verne  
Amiens, France  
gael.le.mahec@u-picardie.fr

Abstract—The increasing number of published data has allowed the development of data mining, resting on the use of the data to extract knowledge. At the same time, to tackle privacy concerns, anonymization models such as $k$-anonymity have emerged. Because $k$-anonymity transforms original data, there is an impact on the utility of altered data for data mining. In this paper, we propose a new writing of the anonymous tables using an anonymization post-treatment. The proposed representation allows to keep more information on the distribution of the original values in the anonymous equivalence classes while being usable directly as input for neural networks for data mining purposes. We test our experimental protocol on two data sets from anonymization research field: A Adult data set and an extract from the register of voters of Florida (USA). With these experiments, we show the superiority in data utility of our approach against classical approaches.

I. INTRODUCTION

The availability of more and more data sets, that are more and more complete, specific and voluminous is a tremendous opportunity for data mining and knowledge discovery for many cultural/scientific researches and business activities.

However, with the availability of personal data emerged major privacy concerns. To share data in accordance with privacy protection laws and the acceptability of citizens for this sharing is a new challenge.

There are many works on Privacy-Preserving Data Publishing (PPDP) and the particular problem of data mining on such data (Privacy-Preserving Data Mining or PPDM). Among them we can mention $k$-anonymity [1], $l$-diversity [2], $t$-closeness [3] or $\epsilon$-differential privacy [4] as the most “famous” approaches. They all have advantages and drawbacks for the particular problem of data mining and knowledge discovery. Reader can refer to [5] and [6] for good analyses of these concepts regarding to the PPDP and PPDM problems. In this paper we focused on the $k$-anonymity concept and particularly the categorical data representation for a use with a machine learning classifier (a Multi-Layer Perceptron classifier). Indeed, obtaining a $k$-anonymous version of a table implies most of the time to alter the data, reducing their utility. The larger is $k$, the more the data are altered to fit with the $k$-anonymity constraints. We studied how a $k$-anonymous data representation can mitigate this information loss and preserve data utility without reducing privacy compared to classical categorical data representations for MLP.

To evaluate consequences for data mining of the data alteration after $k$-anonymization, we used publicly available tables to train different MLP classifiers. We used different table representations of the $k$-anonymous tables as input for the training of the classifiers, resulting in different output classifiers. We showed that a new representation of the $k$-anonymous data significantly improves the performance of the trained models.

The next section presents $k$-anonymity, how to measure “quality” of an anonymous table and how to use such a table as input of a data mining algorithm. Section III presents our new data representation of $k$-anonymous tables for data mining and we discuss the consequences on privacy. Section IV presents the experiments that validate the new representation and compare performance with other data representations. We conclude in Section V.

II. $k$-ANONYMIZATION OF A TABLE

To preserve privacy in the published tables, it is necessary to beforehand anonymize the table. Nevertheless, two kinds of disclosure could appear in anonymous tables [7]: i) a table has identity disclosure if an individual can be uniquely link to a record ; ii) a table has attribute disclosure if new information can be deduced about individuals thanks to the table.

To provide a protection against identity disclosure, $k$-anonymity could be applied to the table [1]. A table is $k$-anonymous if each record is indistinguishable from at least $k – 1$ other records w.r.t a particular set of attributes. To achieve $k$-anonymity, the attributes of a table are classified into the following three categories. Identifier attributes are unique
identifier between an individual and a record of the table (e.g. name, SSN). These attributes have to be removed or replaced by random IDs: this process is called pseudonymization. Quasi-identifiers are not unique to each individual and can not reveal information if they are taken separately. However, they can be used to disclose private information. Sweeney proved in [8] that it is possible to link an individual to a record using quasi-identifiers and other public tables. An equivalence class is a set of records in which the quasi-identifier attributes’ values are all the same. Finally, sensitive attributes are the purpose of the table publication. They are personal information about individuals that publisher has to protect (e.g. income, disease). Publishers have to ensure that the sensitive attributes are not altered but cannot be linked to an individual.

$k$-anonymity deals with quasi-identifier attributes but does not take into account sensitive attributes. Other models, such as $l$-diversity [2] and $t$-closeness [3], ensure that each equivalence class has a specific distribution of sensitive values to limit attribute disclosure. In our context, we focus on $k$-anonymity so as not to influence sensitive values’ distribution.

A. Generalization technique and $k$-anonymization algorithms

To achieve $k$-anonymity, generalization is a common technique. For each quasi-identifier attribute, a generalization hierarchy [1] is built. It translates semantic groupings between values. At level 0 of the hierarchy, leaves are values in the original table (without generalization). Then, the higher a value is in the tree, the most generalized it is.

During a $k$-anonymization, values of quasi-identifiers are replaced by generalized values according to the generalization hierarchies until all equivalence classes are at least $k$ in size. In other words, we merge equivalence classes, such that their quasi-identifiers attributes are identical generalization of their original values. Example 1 presents a 3-anonymization of a table (Table Ia). To define the merging of two equivalence classes, we use the notion of Lowest Common Ancestor LCA. According to Bender in [9], the LCA of two nodes is the ancestor of the two nodes that is located farthest from the root (i.e. the less generalized value in our context). Let $T$ be a table and $Q = \{Q_1, \ldots, Q_m\}$ be its set of quasi-identifier attributes. Let $C$ and $C'$ be two equivalence classes of $T$ whose representatives are $[x_1, \ldots, x_m]$ and $[y_1, \ldots, y_m]$. The merging of $C$ and $C'$ is defined as: $\text{Merge}(C, C') = [\text{LCA}(x_1, y_1), \ldots, \text{LCA}(x_m, y_m)]$. To find an optimal $k$-anonymization is then to find a set of merging operations that minimizes the information loss such that all the equivalence classes have a size of at least $k$.

Example 1. Let consider the table $T$ in Table Ia with 2 quasi-identifier attributes. The first one is Gender ($G$), the second is Race. Figure 1 presents a generalization hierarchy of the Race attribute. Gender attribute can only be generalized to “unknown” represented by “*”. A 3-anonymous version of $T$ is presented in Table Ib. The first three lines of $T$ have been merged in [* mammals] and the other three lines have been generalized in [* cetaceans].

Finding an optimal partition of a table that respects $k$-anonymity is NP-hard [10]. In the literature, we find many frameworks to achieve $k$-anonymity. Some of them, as Incognito of Lefevre et al. [11], do single dimensional global recoding: identical lines in the original table will have the same generalization in the anonymous table (global recoding) and all the values of a quasi-identifier attribute are generalized at the same level of the generalization hierarchy (single dimensional).

On the contrary, algorithms as KACA [12], Greedy $k$-Anonymization Algorithm [13] or clustering-based algorithms such that $k$-member [14] or OKA [15] do multi-dimensional recoding: different levels of generalization can be applied to the values of an attribute.

B. Quality of an anonymous table

There is not uniqueness of the $k$-anonymous table produced using the generalization technique. Among them, the records are more or less altered such that they can be grouped in equivalence classes of size $\geq k$.

An approach to estimate the quality of an anonymous table is to refer to data alteration. In this case an information loss metric is used. It computes the amount of loss of information between a table and an anonymous version of it. It exists many information loss metrics in the literature such that Discernability Metric in [16], Normalized Certainty Penalty in [17], Distortion in [12] or Normalized Lost Leaves Metric in [18].

Another approach consists in using a data mining operation to assess the quality of the anonymous table. In this case, original and anonymous data are consecutively used as input to train models for a given prediction task. Comparing prediction results accuracy of the created models is a way to assess data utility of both original and anonymous data. This practical approach have been used in [19] and [20]. These studies have shown that anonymization does not always come with a drop in data utility. We choose this approach in this paper.

<table>
<thead>
<tr>
<th>Gender</th>
<th>Race</th>
<th>Gender</th>
<th>Race</th>
</tr>
</thead>
<tbody>
<tr>
<td>l1</td>
<td>M</td>
<td>l1</td>
<td>mammals</td>
</tr>
<tr>
<td>l2</td>
<td>F</td>
<td>l2</td>
<td>mammals</td>
</tr>
<tr>
<td>l3</td>
<td>F</td>
<td>l3</td>
<td>mammals</td>
</tr>
<tr>
<td>l4</td>
<td>M</td>
<td>l4</td>
<td>cetaceans</td>
</tr>
<tr>
<td>l5</td>
<td>M</td>
<td>l5</td>
<td>cetaceans</td>
</tr>
<tr>
<td>l6</td>
<td>F</td>
<td>l6</td>
<td>cetaceans</td>
</tr>
</tbody>
</table>

(a) $T$

(b) $T_{ano}$

TABLE I: A table $T$ and a 3-anonymous version of it $T_{ano}$
C. Data mining and anonymous data representations

Data mining algorithms are regularly used such as KNN [21], decision trees [22] or SVM [23]. Multi-Layer Perceptron (MLP) has also been successfully used to perform data mining for years and continues to yield good performances in several domains [24], [25]. We decide to work with an MLP classifier.

MLP takes numerical data as input. Thus categorical data have to be preprocessed to be input and processed. Simple categorical data is usually encoded as a vector with all but one zero element. Regarding an anonymous table, the representing vector will have a size equal to the number of nodes in the hierarchy because a value can be replaced by any element of the generalization hierarchy. Knowing the hierarchy, 3 ways to handle anonymous categorical data are given in [26]:

1) oneClass: Each generalization is an independent category (see Table IIa as an example).
2) fillParent: Any value exhibits all features of its parents (see Table IIb as an example).
3) fillChild: Generalized value exhibits features of all child nodes (see Table IIc as an example)

The first representation is a classical one hot vector representation while the latter two add information from the knowledge of the hierarchy. Those representations will be used as comparison baselines with our proposed representation method.

III. ANONYMOUS DATA REPRESENTATION

To our knowledge, in the case of $k$-anonymization, the data usage framework is the following: dataholders perform the anonymization process on the table (Section II-A) and then publish it. To perform data mining, researchers have to preprocess the table (Section II-C). This framework is illustrated in Figure 2a.

During the anonymization process, information about distribution of values is lost during the merging of equivalence classes. For instance, using the generalization hierarchy in Figure 1, if “Felidae” is found in a record of the anonymous table, we know for sure this record is either a cat, a lion or a tiger but we have no indication on the probability the value is “cat”, “lion” or “tiger”. The preprocessing phase seen in Section II-C tries to take advantage of the hierarchy to conserve data utility before data mining. Preprocessing is thus used to hold back data utility altered by the generalization.

We propose an alternative framework illustrated in Figure 2b. This alternative framework is based on a new data proportional representation. We propose a postprocessing phase after $k$-anonymization that adds the proportions of original values into the equivalence classes for each quasi-identifier before the publication. Each quasi-identifier is then replaced by a vector of its probabilities of having the possible values. Because the proposed representation includes proportions of each value in the equivalence class, it replaces further preprocessing techniques before data mining.

### A. Our proportional representation

We now present the new representation of the anonymous tables that preserves probabilities on the values. Let $T = \{l_1, ..., l_n\}$ be a table and $Q$ be its set of quasi-identifier attributes. For each, $Q \in Q$, we denote by $G_Q$ its generalization hierarchy. Let $T_{ano}$ be a $k$-anonymous version of $T$ obtained with the generalization method. We denote by $C(T_{ano})$ the set of equivalence classes of $T_{ano}$ in which each $C$ is of the form $C = \{l_j \in T_{ano}, j = c_1, ..., c_{|C|}\}$ ($C$ is a subset of lines of $T_{ano}$).

We create a matrix $M$ such that its lines are indexed by the lines of $T_{ano}$, $\{l_1, ..., l_n\}$. Its columns are indexed by the values of the generalization hierarchies. These indices are of the form $Q_{val}$ with $Q \in Q$ and $val$ a value in the generalization hierarchy of $Q$.

For a line $l \in T_{ano}$, a quasi-identifier attribute $Q \in Q$ and a value of the generalization hierarchy of $Q_{val}$, we define the entry $(l, Q_{val})$ of the matrix $M$ as:

$$M(l, Q_{val}) = \frac{|S(l, Q_{val})|}{|C|},$$

with $C$ the equivalence class of $T_{ano}$ such that $l \in C$ and $S(l, Q_{val}) = \{values \ of \ the \ lines \ of \ C \ in \ T \ that \ are \ equal \ or \ can \ be \ generalized \ in \ val \ in \ G_Q\}$. Example 2 presents such a construction of a matrix.

**Example 2.** Let’s consider again the table $T$ in Table Ia. We consider the same generalization hierarchies as in Example 1. We 3-anonymize $T$ to obtain a table $T_{ano}$ with two equivalence classes: $C_1 = \{l_1, l_2, l_3\}$ and $C_2 = \{l_4, l_5, l_6\}$ (see Table Ib). We transform $T_{ano}$ in the matrix $M$ of Figure III. For instance, to fill the entry $M(l_1, R_{Felidae})$ of $M$, we apply the formula in (1). We first look for the equivalence class of $T_{ano}$ in which $l_1$ is: this is $C_1 = \{l_1, l_2, l_3\}$. Then, we get the values of the lines of $C_1$ for the attribute $R$ in the original table $T$: this is $\{cat, lion, dog\}$. We determine the set of values in the previous set that are equal or can be generalized in felidae in the generalization hierarchy of $R$: $S(R, felidae) = \{cat, lion\}$. Indeed, cat and lion are two values that can be generalized in felidae but dog can not. Finally, we obtain $M(l_1, R_{Felidae}) = \frac{|S(R, felidae)|}{|C_1|} = \frac{2}{3}$. That means that the value of $l_1$ for the attribute $R$ has probability $\frac{2}{3}$ to be a felidae.
(a) oneClass: generalizations are independent categories

<table>
<thead>
<tr>
<th>Race</th>
<th>R_{00}</th>
<th>R_{01}</th>
<th>R_{02}</th>
<th>R_{03}</th>
<th>R_{10}</th>
<th>R_{11}</th>
<th>R_{12}</th>
<th>R_{20}</th>
</tr>
</thead>
<tbody>
<tr>
<td>cat</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>dog</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>felidae</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>canine</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>mammals</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

(b) fillParent: any value exhibits all features of its parents

<table>
<thead>
<tr>
<th>Race</th>
<th>R_{00}</th>
<th>R_{01}</th>
<th>R_{02}</th>
<th>R_{03}</th>
<th>R_{10}</th>
<th>R_{11}</th>
<th>R_{12}</th>
<th>R_{20}</th>
</tr>
</thead>
<tbody>
<tr>
<td>cat</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>dog</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>felidae</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>canine</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>mammals</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

(c) fillChild: generalized value exhibits features of all child nodes

TABLE II: Data representations of anonymous data for datamining

<table>
<thead>
<tr>
<th>G_{00}</th>
<th>G_{01}</th>
<th>G_{02}</th>
<th>R_{00}</th>
<th>R_{01}</th>
<th>R_{02}</th>
<th>R_{03}</th>
<th>R_{04}</th>
<th>R_{05}</th>
<th>R_{06}</th>
<th>R_{10}</th>
<th>R_{11}</th>
<th>R_{12}</th>
<th>R_{20}</th>
</tr>
</thead>
<tbody>
<tr>
<td>{1}</td>
<td>0.33</td>
<td>0.66</td>
<td>1</td>
<td>0.33</td>
<td>0.33</td>
<td>0</td>
<td>0.33</td>
<td>0</td>
<td>0.33</td>
<td>0</td>
<td>0.33</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>{2}</td>
<td>0.33</td>
<td>0.66</td>
<td>1</td>
<td>0.33</td>
<td>0.33</td>
<td>0</td>
<td>0.33</td>
<td>0</td>
<td>0.33</td>
<td>0</td>
<td>0.33</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>{3}</td>
<td>0.33</td>
<td>0.66</td>
<td>1</td>
<td>0.33</td>
<td>0.33</td>
<td>0</td>
<td>0.33</td>
<td>0</td>
<td>0.33</td>
<td>0</td>
<td>0.33</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>{4}</td>
<td>0.66</td>
<td>0.33</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.33</td>
<td>0.66</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>{5}</td>
<td>0.66</td>
<td>0.33</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.33</td>
<td>0.66</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

TABLE III: A proportional representation of a 3-anonymous table

2) Florida voters data set: We randomly extract 30162 lines from the register of voters for the state of Florida [28]. We conserve 4 columns as quasi-identifiers: Zipcode (the 5 first digits, 1165 nodes), Gender (only Female and Male, 3 nodes), Race (9 nodes) and Year of Birth (106 nodes). As sensitive attribute, we choose Party affiliation (10 possible values).

B. Anonymization protocol

In our study, we work with Algorithm 1 which is the Greedy k-Anonymization Algorithm presented in [13]. It is a simple k-anonymization algorithm in which an information loss metric is used to guide the equivalence classes mergings (see Section II-A).

Algorithm 1 Greedy k-Anonymization Algorithm

Input: A table $T$, an integer $k$, an information loss metric $\mu$

Output: A $k$-anonymous version of $T$

1: while $T$ is not $k$-anonymous do
2: Choose one of the smallest class $C_{\text{small}}$ in $C(T)$ that is not $k$-anonymous
3: Find a class $C$ in $C(T) \setminus C_{\text{small}}$ such that $\text{Merge}(C_{\text{small}}, C)$ minimizes cost for $\mu$
4: $C(T) \leftarrow C(T) \setminus \{C_{\text{small}}, C\} \cup \text{Merge}(C_{\text{small}}, C)$
5: end while
6: return $T$

Based on the metrics comparison exposed in [18], we choose Normalized Lost Leaves Metric as information loss metric in the algorithm.

Thanks to the Greedy k-Anonymization Algorithm, we produce $k$-anonymous versions of the two data tables for 14 values of $k$ between 3 and 15000. Please note that 15000 is not a realistic value for the $k$-anonymity of a 30162 lines but it allows us to evaluate the behaviour of our algorithms. Because the range of values is large, we use a logarithmic scale on abscissa representing $k$ in Section IV-D.

C. Training protocol

The training of the MLP classifiers has been conducted using the scikit-learn library [29]. Because our goal is only

\[ k \in \{3, 4, 5, 10, 20, 50, 100, 250, 500, 1000, 2000, 5000, 10000, 15000\} \]
to compare data representation, the number of hidden layers, their size and all hyper parameters are arbitrarily fixed. Each model have the same architecture of 2 hidden layers counting respectively 5 and 2 neurons. The activation function is “reLu”. The training is performed using Adam solver with a constant learning rate of 1e-3. The batch size is set to 200. The training is stopped if there is no evolution on loss after 10 epochs with a maximum epoch number of 500. Two third of the tables records were used as the training set while the rest was used as validation set.

Several metrics exist to assess the quality of the model. For Adult, because the target attribute is binary, we have chosen to use the area under the Receiver Operating Characteristics curve (AUC-ROC). For Florida, we have chosen the accuracy of the model because of the 10 possible values for the sensitive attribute. These metrics allow us to evaluate the global performance of a model with a scalar.

The models are trained and evaluated using different representations of the same anonymous table. For reproducibility, each training has been repeated 10 times with a different random seed for initialization, leading to 10 different models per training phase. The mean and standard deviation of the 10 evaluation values produced are used to compare data utility of the data representations.

D. Results

For Figures 3 to 5, blue curve is our proportional representation; orange, green an red curves represent results for respectively fillParent, oneClass and fillChild representations.

1) Training on the non-anonymous table: Let us first consider a full access to the non-anonymous tables. For \( k = 1 \) (non-anonymous table), proportional and fillParent representations are the same. For both Adult and Florida, 10 models are trained on the non-anonymous data using this proportional (or fillParent) representation. Figure 3 represents the mean performance of MLP classifiers when tested on anonymous tables. Each point on a curve is the mean of the values obtained with the 10 trained classifiers with different seeds. For readable reasons, standard deviation is shown on the figures only for the proportional representation.

For both tables, we note a decrease in prediction performance when \( k \) increases. This effect is common to all representations. This is logical and expected as an increase in \( k \) produces a more altered table with numerous similar records. When \( k \)-value exceeds 5000, any representation maintains a consistent AUC-ROC value greater than 0.5 for Adult, reflecting the inability to keep any data utility.

We note our proportional representation consistently outperforms every representation for each tested \( k \). We argue our representation avoids some data loss, thus increasing the later data utility.

2) Training on a \( k \)-anonymous table: The access to the full non-anonymous table is not always possible. In this case, the training has to be done using \( k \)-anonymous table. In Figures 4 and 5, the prediction models are trained with a 100-anonymous table. Each sub-figure accounts for a different data representation used as input for the models training. In each sub-figure, each curve accounts for a different data representation used as input for the models evaluation. The color code for the curves is the same as in Figure 3.

Let us first compare data representations for evaluation (curves inside each sub-figure). We note the prediction is consistently better when using proportional representation for evaluation, independently from the data representation used for training. We believe that our proposed data representation allows to keep useful information as \( k \) increases. We also note the second best results are achieved by fillParent representation. In the case a predicting model is already trained, we believe our data representation should be used if possible.

Let us now compare the data representations for training (results between sub-figures). The highest predictive values are obtained evaluating with proportional representation on models trained with fillParent or proportional representations. When testing with fillParent representation (2nd best results), results seems best with the models trained with fillParent representation. Since evaluating with proportional yields the same results independently from training with fillParent or proportional, if one doesn’t know what data representation will be used for future predictions, we tend to recommend a training with the fillParent representation.

Comparing Figure 3 and Figures 4 and 5, we note that
Fig. 4: Adult, AUC-ROC, trained on a 100-anonymous table

(a) model trained with proportional representation

(b) model trained with fillParent representation

(c) model trained with oneClass representation

(d) model trained with fillChild representation

Fig. 5: Florida, accuracy, trained on a 100-anonymous table

(a) model trained with proportional representation

(b) model trained with fillParent representation

(c) model trained with oneClass representation

(d) model trained with fillChild representation
models trained with non-anonymous tables and models trained with 100-anonymous tables performs at the same level. This confirms the findings of [20], that a $k$-anonymous table doesn’t always come with a drop in terms of data utility.

Overall fillParent and proportional representations outperform oneClass and fillChild representations. The latter two might not be indicated to be used in practice. In our experiments, using these representations as training to evaluate other representations yields irrelevant results. The opposite is also true. We argue the fillChild representation is too different from other representations while oneClass representation erases semantic groupings between records of a table.

V. Conclusion and Discussion

In this paper, we have introduced a novel way to represent $k$-anonymous tables that preserves $k$-anonymity while keeping more information than a regular $k$-anonymization. This new writing contains distribution of the data inside the equivalence classes. It can be implemented after the anonymization process and before data release. With experiments on two data sets, we show our representation of $k$-anonymous tables allows better predicting capabilities for already trained MLP models.

While respecting $k$-anonymity, the proposed table representation nevertheless introduces a bit more of attribute disclosure. It could be useful in a future work to establish the impact of this representation on other privacy models such as $l$-diversity or $t$-closeness. We could also measure the impact of this representation in case of an adversarial attack. This paper limits its experiments to the use of an MLP classifier as a way to assess the utility of data. We believe our method could be used with other data mining classification algorithms like K-nearest neighbors or decision trees to measure the effect of our data representation on other classification algorithms.

References