A Baseline for Attribute Disclosure Risk in Synthetic Data

Markus Hittmeir
SBA Research
Vienna, Austria
mhittmeir@sba-research.org

Rudolf Mayer
SBA Research
Vienna, Austria
rmayer@sba-research.org

Andreas Ekelhart
SBA Research
Vienna, Austria
aekelhart@sba-research.org

ABSTRACT

The generation of synthetic data is widely considered as viable method for alleviating privacy concerns and for reducing identification and attribute disclosure risk in micro-data. The records in a synthetic dataset are artificially created and thus do not directly relate to individuals in the original data in terms of a 1-to-1 correspondence. As a result, inferences about said individuals appear to be infeasible and, simultaneously, the utility of the data may be kept at a high level. In this paper, we challenge this belief by interpreting the standard attacker model for attribute disclosure as classification problem. We show how disclosure risk measures presented in recent publications may be compared to or even be reformulated as machine learning classification models. Our overall goal is to empirically analyze attribute disclosure risk in synthetic data and to discuss its close relationship to data utility. Moreover, we improve the baseline for attribute disclosure risk from the attacker's perspective by applying variants of the RadiusNearestNeighbor and the EnsembleVote classifiers.

CCS CONCEPTS

• **Computing methodologies** → **Supervised learning**; • **Security and privacy** → **Data anonymization and sanitization**; *Usability in security and privacy*; *Privacy protections*;

KEYWORDS

Privacy-Preserving Data Mining, Synthetic Data, Machine Learning

ACM Reference Format:
Markus Hittmeir, Rudolf Mayer, and Andreas Ekelhart. 2020. A Baseline for Attribute Disclosure Risk in Synthetic Data. In *Proceedings of the Tenth ACM Conference on Data and Application Security and Privacy (CODASPY '20), March 16–18, 2020, New Orleans, LA, USA.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3374664.3375722

1 INTRODUCTION

The technological advances of recent years led to an increase in the collection and storage of large amounts of data. Micro-data, i.e. data that contains information about e.g. individuals, is collected in domains such as health care, employment or social media. Similarly, there has been an increase in the capability and the interest to analyse data. Its release and distribution, however, bares the risk of compromising the confidentiality of sensitive information and the privacy of affected individuals. To comply with ethical and legal standards such as the EU's General Directive on Data Protection (GDPR), data holders and data providers have to take measures to prevent attackers from learning sensitive information from the released data, often referred to as *statistical disclosure control* (SDC).

In the case of micro-data, two possibilities of disclosure of sensitive information are widely considered. *Identification disclosure* happens when an adversary is able to conclude that a certain record in the dataset belongs to a certain individual. *Attribute disclosure* happens whenever the dataset allows the attacker to learn new information about a specific individual in question, e.g. the value of a certain attribute. Identification disclosure often leads to attribute disclosure, as every attacker's ultimate goal is to gain information on their victim. However, attribute disclosure can also happen without the attacker uniquely identifying the record of their victim in the dataset, e.g. by the matching techniques discussed in this paper.

In most cases, it does not suffice to remove directly identifying attributes (primary identifiers), such as names or social security numbers, from the data. To minimize disclosure risks, approaches like Differential Privacy [1] and k-Anonymity [14] have been developed. The reader may consult the survey [3] for a general overview on traditional privacy-preserving data publishing methods.

In this paper, we will consider an alternative disclosure control measure, namely the generation of synthetic data. One of the first applications is described by Rubin in [11], where multiple imputation is used to synthetically generate certain columns of datasets. An overview on more than 20 different scenarios is given in [12]. An evaluation of the utility of synthetic data, generated by various tools, for supervised machine learning tasks, specifically classification tasks, is also given in [4].

In our experiment in Section 4, we use three recently published synthetic data generation tools: The *Synthetic Data Vault* has been developed in 2016 by N. Patki et al. at MIT, and is implemented in Python. It builds a model based on estimates for the distributions of each column. In order to preserve the correlation between attributes, the synthesizer applies a multivariate version of the Gaussian copula and, subsequently, computes the covariance matrix. For more details and an utility evaluation conducted by the developers, the reader may consult the original publication ([7]).

The second tool we use is the *DataSynthesizer*, proposed in 2017 by H. Ping et al. and also implemented in Python. The user is able to specify one of three modes, namely 'random mode', 'independent

CODASPY '20, March 16–18, 2020, New Orleans, LA, USA

ACM ISBN 978-1-4503-7107-0/20/03…$15.00
https://doi.org/10.1145/3374664.3375722

attribute mode', or 'correlated attribute mode'. If the tool should preserve dependencies between the attributes, the last mode should be chosen. The tool then generates synthetic data based on a Bayesian network model learned from the original data. For extended SDC, DataSynthesizer uses the framework of Differential Privacy and offers the possibility to determine the amount of injected noise. More information on this method can be found in [8].

Finally, we use the *synthpop* [6] package for *R*, which has been created by B. Nowok et al. at the University of Edinburgh. Here, the default synthesis method is a CART (Classification And Regression Tree) algorithm. However, the user is able to specify a large number of parameters. Synthpop also contains a function for SDC[1], which may be applied to the resulting synthetic dataset.

Usually, a distinction is made between fully and partially synthetic data. Fully synthetic data means that the whole dataset is synthesized, whereas partially synthetic data contains a mixture of synthesized values for sensitive and original values for nonsensitive attributes. In 2009, Reiter and Mitra [9] proposed identification disclosure risk estimations for partially synthetic data. In this paper, we consider attribute disclosure risks on fully synthetic data. The notion of identification disclosure is not in our focus, since fully synthetic records do not relate to original records in terms of a 1-to-1 correspondence. However, this does not exclude the possibility of attribute disclosure, for which it is supposed that the attacker knows the values of certain attributes of their victim (called the *key* variables) and wants to learn the value of some sensitive attribute (called the *target* variable). Approaches for measuring the related risk have been proposed by Reiter et al. [10] and by Taub et al. [15]. The methods differ by the amount of the assumed background knowledge $\mathcal{B} = \{A, S\}$ of the attacker. A denotes the attacker's knowledge about records in the original (unsynthesized) dataset, and S comprises available information about the process of generating the synthetic data, like code for the synthesizer or a description of the used tools. Reiter et al.'s approach assumes a worst case attacker scenario, in which the adversary knows all entries in the original dataset except the target attribute value they want to learn. While the authors admit that this assumption may be viewed as overly conservative and unrealistic, they suggested that their measures offer a type of upper bound on the disclosure risks. Taub et al.'s approach, on the other hand, assume an attacker's behavior that does not rely on $\mathcal{B}$ at all, and is therefore feasible for $A = S = \emptyset$. The related research question asks for a baseline, for a lower bound on the attribute disclosure risk: given only the synthetic dataset and the values of certain key attributes, which procedures are always available to the attacker that may help him to learn the value of a certain target attribute? This question is of great importance for analyzing the general usefulness of data synthesis as privacy-preserving method.

Our main contribution is the generalization of Taub et al.'s approach, which is based on the concept of *Correct Attribution Probability*. The technique finds those records in the synthetic dataset which match a certain combination of key variables. For example, the attacker may know that this set of values belongs to a certain individual in the original data. For the found synthetic records, the distribution of the value of the target attribute is computed, which allows to assign a risk probability for the exposure of the real value of the corresponding individual in the original dataset. However, it may happen that the distinct combination of key attribute values of some row in the original data does not occur in the synthetic data. While the original approach either ignores such non-matches or assigns probability 0, our generalization allows to extend the risk analysis to these records. In our evaluation, we demonstrate the merit of this approach and compare it to machine learning classifiers which the attacker might use to extract information from the data and obtain a prediction for the target variable of their victim.

The mentioned approaches exploit global, not local properties of the dataset. While arguments have been brought forward that for an attacker there is little additional knowledge to be gained from synthetic data that describes publicly well known correlations in data, we want to stress that the task of estimating attribute disclosure on fully synthetic data (or on corresponding models) is particularly relevant whenever the comprised information and the correlations in the original data are **not** publicly known. This is often the case for data about sub-populations and for business data. In general, our evaluation shows that the attacker is able to gain knowledge from the synthetic data that increases the accuracy of their predictions.

The remainder of this paper is structured as follows: In Section 2, we discuss related work and, on this basis, the relation between data utility and attribute disclosure risks by considering the attacker's situation as classification problem. In Section 3, we improve the baseline for attribute disclosure risk by generalizing the approach established in [15]. In Section 4, we evaluate our approach and compare the performance of several machine learning models on the attacker's classification problem. Finally, in Section 5, we will draw our conclusions and describe ideas for future work.

2 ATTACKER'S CLASSIFICATION PROBLEM

It has already been mentioned that, for fully synthetic data, the notion of identification disclosure is not clear cut. From an attacker's perspective, the approach to gain information by linking certain synthetic records to individuals is not promising, as such links generally do not exist. Attribute disclosure, on the other hand, does not necessarily depend on such linkages. There are other ways to use data and prior knowledge for learning about a sensitive target attribute value, one of which will be analysed in the next section. Still, it seems highly unlikely that synthetic data can ever be used by the adversary to infer information with absolute certainty. In order to see this, assume that one of the records in the Adult Census Income dataset[2] belongs to our neighbor. Our prior knowledge consists of the values of the key attributes 'age', 'gender', 'race', 'occupation', 'marital-status' and 'native-country'. We are nosy and want to know if she earns more or less than $50K a year. Consequently, 'income' is our target attribute. We simply search for her combination of key attributes and find only one record in the whole dataset that matches all these values. We can be certain that this is the record of our neighbor, and may obtain the respective target

[1] https://rdrr.io/cran/synthpop/man/sdc.html

[2] https://archive.ics.uci.edu/ml/datasets/census+income

value. Studies such as [13] have shown that with similar (actually fewer) attributes, a large majority of 87% of the US-residents can be identified, so this is a likely scenario. Even if we find more than one record with this combination, we may still be able to draw certain conclusions in the case where all of them have the same 'income' value (a situation that the concept of *l-diversity* [5] would address). If, however, we do *not* have the original dataset at hand, but just a synthesized version, the situation is quite different. It now can happen that no record comprises the values of our known key attributes. Even if there is a single record matching our prior knowledge, we cannot be certain about the entry of the 'income' attribute. As a result of the data synthesis, this target value - as well as the values of the other attributes not known by us - might deviate from our neighbor's real entries. We again face the same difficulties we already discussed in the context of identification disclosure: the record in question is not our neighbor's, nor is it our neighbor's synthesized record. In most cases, it is the product of randomized draws from a model described by global, not local, properties of the dataset.

In accordance with these considerations, attribute disclosure risk in synthetic data is measured by providing *probabilities* for the exposure of the real target value of records in the original data. We give an example by discussing Reiter et al.'s [10] already mentioned approach. Let $D = \{(x_i, y_i) : i = 1, \ldots, n\}$ be the matrix comprising the original database, where x_i is the vector of the i-th record's values of non-sensitive attributes, and y_i is the vector of the i-th record's values of sensitive attributes which are subject to synthesis. Note that, for fully synthetic data, $X = (x_i : i = 1, \ldots, n)$ is empty. By $Z = (Z^{(1)}, \ldots, Z^{(m)})$, we denote the m synthetic datasets generated by the data provider. Assume that an attacker wants to learn the vector y_i for some record i in D. Let $\mathcal{B} = \{A, S\}$ be the background knowledge of the attacker. We recall that A consists of information about the original data and, for Reiter et al.'s approach, is set to $A = \{\{(x_j, y_j), \text{ for } j \neq i\} \cup x_i\}$. Hence, it is assumed that the attacker knows the complete original dataset except the target value(s) of interest. Furthermore, we recall that S comprises knowledge about the synthesizer. Finally, let Y_i denote the random variable representing the attacker's uncertain knowledge of y_i. The sample space of Y_i is given by all possible values of y_i in the population. For evaluating a guess y^* for y_i, Reiter et al. assume that the attacker seeks the Bayesian posterior distribution $\mathrm{P}(Y_i = y^* \mid Z, X, A, S)$ which, for a discrete random variable Y_i, is equal to

$$\frac{\mathrm{P}(Z \mid Y_i = y^*, X, A, S)\,\mathrm{P}(Y_i = y^* \mid X, A, S)}{\sum_y \mathrm{P}(Z \mid Y_i = y, X, A, S)\,\mathrm{P}(Y_i = y \mid X, A, S)},$$

where the sum in the denominator is taken over all possible values y of y_i in the population. Depending on the circumstances, a variety of techniques are proposed for estimating the prior distribution $\mathrm{P}(Y_i = y^* \mid X, A, S)$ and the probability $\mathrm{P}(Z \mid Y_i = y^*, X, A, S)$ of generating Z. For the first, one may either use a discrete uniform distribution or assume an adversary that already uses A to form prior beliefs. For the latter, importance sampling techniques are adopted and coupled with Monte Carlo simulation.

Based on the resulting value $\mathrm{P}(Y_i = y^* \mid Z, X, A, S)$, the data provider is able to compute several risk measures for the released synthetic dataset(s). One option mentioned by Reiter et al. is to compute

$$R_i = [\mathrm{argmax}_y \,\mathrm{P}(Y_i = y \mid Z, X, A, S) = y_i], \tag{2.1}$$

where

$$[p] = \begin{cases} 1 & \text{if } p \text{ is true,} \\ 0 & \text{otherwise,} \end{cases}$$

is the so called Iverson bracket. Subsequently, one may want to evaluate the disclosure risk of the complete dataset by deciding whether $R = \sum_{i=1}^{n} R_i/n$ is acceptably low. Another option would be to compare $\mathrm{P}(Y_i = y^* \mid Z, X, A, S)$ to the prior belief, e.g. by considering the multiplicative increase.

Correct Attribution Probability ([15]), an idea discussed in Section 3, is rather different from the Bayesian Estimate described above. As mentioned in the introduction, no background knowledge $\mathcal{B}$ of the attacker is assumed. Clearly, this also restricts their possibilities of privacy violations. As a result of computing the disclosure risk measure, however, the data provider also obtains a distribution of all possible values of y_i and may use the related percentage scores in the same way as $\mathrm{P}(Y_i = y^* \mid Z, X, A, S)$ to evaluate the overall disclosure risk. We may conclude that, from the attacker's perspective, both approaches provide means to solve the following task.

Attacker's Classification Problem: *Given some background knowledge $\mathcal{B}$, the synthetic dataset(s) Z and the values of key attributes of some record in the original dataset, obtain a prediction on the target value of said record.*

The goal of this paper is to discuss the possibilities of the attacker to approach even the most restricted scenario of this problem, that is, when they have no background knowledge and only a single published synthetic dataset at hand. The purpose of this viewpoint is to establish a baseline, a set of tools for privacy invasion that is available to the adversary under all circumstances and that, on the flip side, should be always taken into consideration by data holders and data providers.

Clearly, machine learning models for classification are part of the attacker's toolkit, as these are directly applicable to the discussed situation. The question is: how well do models that are trained on the synthetic data perform, if they are applied back on the original data? Notably, a very similar question is often discussed in the context of the *utility* of the synthetic dataset. The answer depends on two factors:

(1) How strong is the correlation between the key attributes and the target attribute?
(2) To which degree does the synthesizer preserve the global properties of the original data, that is, the distributions of attributes and the dependencies between them?

If the correlation between sensitive variables and typical quasi-identifiers is strong in the original data, the only way to reduce disclosure risk is to conceal these dependencies in the synthetic data, e.g. by adding more noise in the process of synthesis. This will result in the loss of information and, hence, in a reduction of the utility of the synthetic dataset. For examples and simulations, we refer the reader to Section 4.

At first glance, the proposed viewpoint might appear counter-intuitive. The information about the sensitive target attribute of the individual in question is not disclosed to the attacker by identifying the corresponding record or using some other local vulnerability of the data, but by considering and exploiting its global properties. However, if any tool available to the attacker results in high probability of exposure of the true target value of certain records, the privacy of affected individuals is clearly violated. Furthermore, for reasons already discussed, the focus on global properties lies in the nature of synthetic data disclosure risk assessment. In order to corroborate our statements, we will now discuss the relation between Correct Attribution Probability scores and one of the less well-known machine learning classifiers, namely the Fixed-Radius Nearest Neighbor search.

3 CORRECT ATTRIBUTION PROBABILITY

The concept of Correct Attribution Probability (CAP) has been introduced in [2] and elaborated on in [15] by J. Taub et al. In the first publication, M. Elliot used CAP to estimate disclosure risks of datasets generated by the synthpop [6] package in *R*, which was developed by the SYLLS Team at the University of Edinburgh. For assessing attribute disclosure risk, CAP assumes that the attacker knows the values of a set of key attributes for an individual in the original dataset, and wants to learn the respective value of some target attribute. CAP measures the disclosure risk of the individual's real target value in the case where the adversary has access to the synthetic dataset. In [15], the method is presented for a situation where the attributes in the key as well as the target attribute are all categorical. For the remainder of this section, we will keep this assumption. The reader is referred to [2] for a variant handling continuous target variables.

Consider a dataset consisting of micro-data with n records representing individuals and an unspecified number of attributes in the columns. For $j \in \{1, \dots, n\}$, let $K_{o,j}$ be the vector representing the values of the key attributes of the j-th record in the original dataset, and let $T_{o,j}$ be the corresponding value of the target attribute. Similarly, we define $K_{s,j}$ and $T_{s,j}$ for the synthetic dataset. The CAP score for record j in the original dataset is the empirical probability of its target value given its key attribute values, that is

$$\mathrm{CAP}_{o,j} := \mathrm{P}_o(T_{o,j} \mid K_{o,j}) = \frac{\sum_{i=1}^{n}[T_{o,i} = T_{o,j} \wedge K_{o,i} = K_{o,j}]}{\sum_{i=1}^{n}[K_{o,i} = K_{o,j}]}.$$

By indexing the probability $\mathrm{P}_o(\bullet)$, we indicate that our sample space is the original dataset. Additionally, we define the CAP score for the synthetic dataset, that is

$$\mathrm{CAP}_{s,j} := \mathrm{P}_s(T_{o,j} \mid K_{o,j}) = \frac{\sum_{i=1}^{n}[T_{s,i} = T_{o,j} \wedge K_{s,i} = K_{o,j}]}{\sum_{i=1}^{n}[K_{s,i} = K_{o,j}]}.$$

The basic idea is that the attacker is supposed to search for all records in the synthetic dataset that match the key attribute values known by them. This subset of data points is often referred to as *equivalence class* of $K_{o,j}$. Inside this class, they then calculate the distribution of the occurring values of the target attribute. Clearly, $\mathrm{CAP}_{s,j}$ corresponds to the proportion of the actual target value $T_{o,j}$ in this equivalence class. In this sense, $\mathrm{CAP}_{s,j}$ measures the risk of disclosure of this information about the individual represented by the j-th record in the original data. In order to evaluate $\mathrm{CAP}_{s,j}$, the authors of [15] computed the mean value over all the records. Finally, they compared the result to the mean of $\mathrm{CAP}_{o,j}$ as well as to the mean marginal probabilities of $T_{o,j}$ in the original dataset. The authors also noted that $\mathrm{CAP}_{s,j}$ is undefined if the vector $K_{o,j}$ does not occur in the synthetic dataset. In their evaluation and in the calculation of the mean CAP score, they dealt with this scenario in two different ways:

(1) Coding the corresponding CAP scores as 0
(2) Treating the corresponding CAP scores as undefined

We will discuss both options and their justifications in our subsequent analysis of the approach.

It is worth to mention that there is a close relation between CAP scores and the well-known concepts of k-anonymity and l-diversity (see [5, 13]).

k-Anonymity: *A dataset has the k-anonymity property if, for every combination of attributes occurring in the data, the corresponding equivalence class consists of at least k elements.*

(Distinct) l-Diversity: *A dataset has the l-diversity property if, in every equivalence class, the sensitive variable (e.g., the target attribute T) takes on at least l distinct values.*

If we restrict our attention to the original dataset and assume that the k-anonymity property is not satisfied for at least $k = 2$, there are records that, for some key, are the only elements in their equivalence class, and hence there are $j \in \{1, \dots, n\}$ with $\mathrm{CAP}_{o,j} = 1$. If l-diversity is not satisfied for at least $l = 2$ and, therefore, there are not at least l distinct target values in each equivalence class, the same is true. In general, if datasets satisfy l-diversity for higher values of l, the CAP scores of the records are bound to be lower, and vice versa.

In the remainder of this section, we translate the CAP score approach into a solution for the attacker's classification problem. Moreover, we improve the approach from the attacker's perspective. We start by discussing the *Fixed-Radius Nearest Neighbor* classifier (FR-NN), which is implemented in the Python scikit-learn machine learning package *scikit-learn*[3].

Fixed-Radius Nearest Neighbor: *Based on a metric m and a radius r specified by the user, this algorithm classifies data points by implementing a majority vote among neighbors within r.*

This variant of the better known k-Nearest Neighbor classifier is based on an efficient search for neighboring data points, which, depending on the circumstances, may be realized by the BallTree or the KDTree algorithm. In scikit-learn's implementation, the user can also specify a label for outlier samples which do not have neighboring data points within r.

We now reconsider the attacker's approach that is assumed by the CAP disclosure risk measure. For a certain attribute key K, the attacker knows $K_{o,j}$ for some record j in the original data, and has access to the synthetic dataset. The adversary then computes the equivalence class of $K_{o,j}$ in the synthetic dataset and, subsequently,

[3] https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.RadiusNeighborsClassifier.html

the distribution of the target attribute T of interest. Now let $\mathcal{S}$ be the synthetic dataset and $\mathcal{S}\big|_{K,T}$ the dataset that results from omitting all attributes but the target T and those in the key K. Then the attacker's approach is equivalent to conducting a FR-NN classification for $K_{o,j}$ on $\mathcal{S}\big|_{K,T}$. Note that we may choose a variety of metrics without affecting the result of the classification, since the attacker only considers neighbors within $r = 0$ (that is, equal data points). However, given that the approach is based on matches for the attributes in the key K, it makes sense to choose the Hamming Distance for m. For two data points (records) $a = (a_1, \dots a_s)$ and $b = (b_1, \dots, b_s)$, the Hamming Distance is defined as

$$\Delta(a, b) := |\{j \in \{1, \dots, s\} : a_j \neq b_j\}|.$$

As a result of this application, the attacker obtains percentages for the possible values of the target T according to their occurrence in the equivalence class, that is, the $r = 0$ neighborhood. The percentage of the real target value $T_{o,j}$ is equal to $\text{CAP}_{s,j}$. It makes sense to assume that the attacker is interested in both the target value with the highest percentage, that is, the result of classification via FR-NN, as well as in all occurring values together with their percentages.

For several reasons, the discussed approach assumed by the CAP measure is not optimal for solving the attacker's classification problem. For example, it may happen that $K_{o,j}$ does not occur in the synthetic dataset, hence does not have any neighbors within $r = 0$. $\text{CAP}_{s,j}$ is then undefined and, similarly, the FR-NN classifier is not able to assign a label. It has already been mentioned that the authors of [15] dealt with this scenario in two different ways, namely by either coding the corresponding CAP scores as 0 in the calculation of the mean CAP score, or treating them as *undefined*, which means that the respective record does not count towards n. In Section 3.3 of [15], justifications for both options are given. According to these, the basis for assigning a 0 is that a non-match is considered to have zero probability of yielding a correct attribution, whereas the logic behind recording non-matches as undefined is that an adversary is more likely to stop their attempt with a non-match.

Both options of handling the CAP scores correspond, in some way, to the inability of the attacker's FR-NN classifier to provide a label. However, we now propose an alternative method for the attacker to handle a non-match, which will lead to an improvement of the approach from their perspective. Consider the example of the Adult Census dataset from Section 2. We want to learn if our neighbor earns more than \$50K a year. We know that she is in the dataset and we gained access to a synthesized version. Furthermore, we know her age, gender, race, occupation, marital-status and native-country, all of which are attributes in the dataset. A quick search reveals that no record in the synthetic data is a complete match for these attribute key values. Instead of giving up, we can now search for records that match at least 5 of the 6 attributes in the key. If we do find such records, we proceed by calculating the distribution of their target attribute values. If not, we try for records that match at least 4 attributes, and so on. The resulting algorithm may be implemented as follows.

Algorithm 3.1. *Input:* A synthetic data set $\mathcal{S}$, a target attribute T in $\mathcal{S}$ and an attribute key K together with a value vector $K_{o,j}$ of an original data's record

Output: A prediction T^* for $T_{o,j}$

1: Set $N = \emptyset$ and $r = 0$.
2: **while** $N = \emptyset$ **do**
3: $\quad N \leftarrow \left\{a \in \mathcal{S}\big|_{K,T} : \Delta\left(K_{o,j}, a\big|_K\right) = r\right\}$, where $a\big|_K$ omits the value of T
4: $\quad r \leftarrow r + 1$
5: Choose T^* via majority vote among the values of T for the elements of N

Similarly, we may describe this algorithm as repeated application of the FR-NN classifier for $r = 0, 1, 2, \dots$ and the Hamming Distance. We stop as soon as neighbors are found and a label can be assigned. The algorithm may easily be adapted to not only return a prediction T^*, but also the percentages of the possible values for the target attribute T. We believe that this procedure is superior to the approach assumed by the CAP disclosure measure, for the following reasons:

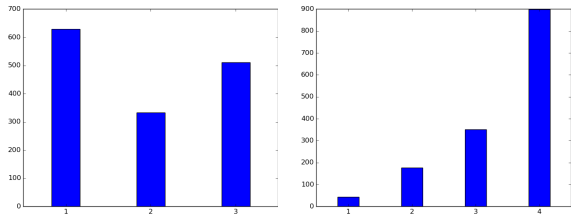
(1) The methods yield the same result for all $K_{o,j}$ that appear in the synthetic data. Only non-matches are handled differently.
(2) Non-matches are more likely to occur for longer attribute keys. However, the attacker is unlikely to stop her attempt to learn sensitive information from the synthetic data because her prior knowledge about the victim is, in this sense, "too detailed". Obtaining a prediction based on smaller attribute keys would be considered better than having no prediction at all. Moreover, the attacker is still able to use all of her prior knowledge by not considering one fixed smaller attribute key, but searching for all records within a certain radius to the vector of known attribute values.
(3) Synthetic data with high utility preserves certain dependencies between attributes and is therefore also likely to yield high accuracy scores for our variant of the FR-NN classifier.

The third reason actually applies to all kind of machine learning classification models. There is nothing special about FR-NN or the general approach to search for matches of the known attribute values. The attacker's classification task may, like any other classification problem, be solved by a variety of different algorithms. We compared the original CAP score approach and our procedure presented above to several algorithms like NaiveBayes, RandomForest and LogisticRegression. For the results of our experiments, we refer the reader to Section 4.

We have now discussed the improved approach from the attacker's perspective. Additionally, we may define a corresponding generalized CAP disclosure risk measure that may be used by the data provider. We therefore conclude this section by extending $\text{CAP}_{s,j}$ to

$$\text{GCAP}_{s,j} := \frac{\sum_{i=1}^{n}[T_{s,i} = T_{o,j} \wedge \Delta(K_{s,i}, K_{o,j}) = \rho]}{\sum_{i=1}^{n}[\Delta(K_{s,i}, K_{o,j}) = \rho]},$$

where $\rho := \min\{r \mid \exists i \in \{1, \dots, n\} : \Delta(K_{s,i}, K_{o,j}) = r\}$. Both notions coincide for $\rho = 0$, but $\text{GCAP}_{s,j}$ is also defined when $\text{CAP}_{s,j}$ is not.



(a) 'contraceptive method' (b) 'education of husband'

Figure 1: Distribution of target variables

4 EVALUATION

In this section, we compare GCAP to CAP and also apply other machine learning algorithms to the attacker's classification problem. We use the Contraceptive Method Choice dataset[4] for our experiment, which is a subset of the 1987 National Indonesia Contraceptive Prevalence Survey. The table consists of 1,473 samples of married women and 10 attributes. It appeared suitable for our purposes because the attributes are an interesting composition of quasi-identifiers and potentially sensitive attributes. Furthermore, all attributes are either categorical or (in the case of 'age') may be treated as such, which has been assumed in the presentation of the approach in the last section.

In our experiment, we analyze the following two scenarios. We consider two subsets of the dataset's attributes as quasi-identifiers, and two different target variables:

(1) QI = {'age', 'education', 'education of husband', 'number of children', 'religion', 'now working?', 'occupation of husband'}
Target = 'contraceptive method'
(2) QI = {'age', 'education', 'number of children', 'religion', 'now working?'}
Target = 'education of husband'

Scenario (1) is based on the fact that the preferred contraceptive method, as well as whether contraception is used at all, is potentially sensitive information for the individuals in the original dataset. The idea behind Scenario (2) is to investigate the possibility of gaining information about the husbands based solely on knowledge about the wives. Note that the attribute 'contraceptive method' has three distinct values in its domain, whereas 'education of husband' has four. Figure 1 shows the distribution of the target attributes in the original dataset.

In order to demonstrate the main difference between GCAP and CAP, we will use a mixture of smaller and larger key sizes. In Scenario (1), we use attribute keys of length three and six. To avoid limiting the analysis to certain subsets of the quasi-identifiers, we considered all subsets of QI with three and six elements. As a result, we investigated a large number of situations. For Scenario (2), we did the same for all attribute keys of length two and four. We will use the same attacker scenarios to discuss the capabilities of other machine learning classifiers, as well as the boundaries of our baseline approach. Let $\mathcal{D}$ be the table of the Contraceptive Method Choice dataset. For both Scenarios (1) and (2) and each key length k, we performed the following procedure.

[4]https://archive.ics.uci.edu/ml/datasets/Contraceptive+Method+Choice

(1) Generate four synthesized versions of $\mathcal{D}$ of equal length:
- The DataSynthesizer without Differential Privacy
- The DataSynthesizer with Differential Privacy ($\varepsilon = 0.1$)
- The Synthetic Data Vault
- The synthpop package in R

(2) Compute all k-element subsets of the quasi identifiers QI of the respective scenario. Each subset corresponds to an attribute key used in the following step.
(3) For each dataset, for each attribute key and the target of the scenario:
- Compute the the CAP scores of all records in $\mathcal{D}$, where non-matches get CAP score 0.
- Compute the CAP scores of all records in $\mathcal{D}$, and ignore non-matches.
- Compute the GCAP scores of all records in $\mathcal{D}$.

Note that non-matches do not occur on the original dataset, hence the notions of GCAP and CAP are equivalent, and the scores match. As part of our experiment, we also want to compare the disclosure risks on the synthetic datasets generated by the different tools. Therefore, we applied all synthesizers with default parameters to avoid any bias or unintended optimization. One exception is the Differential Privacy parameter for demonstrating its effect and the user's possibilities to influence the risk. In our summary of the experiment's results, we included $\varepsilon = 0.1$. Lower values of ε lead to more distortions in the data, whereas setting $\varepsilon = 0$ means to turn of Differential Privacy. Putting $\varepsilon \gg 0.1$, one injects less noise and therefore observes results that are much closer to those on datasets produced by the DataSynthesizer without Differential Privacy. Since these observations agree with the definition of ε-Differential Privacy, we focused on presenting the results of the choice $\varepsilon = 0.1$, which is also used in the tool's documentation.

In Step (3), the scores are computed for each entry in $\mathcal{D}$. As discussed in Section 2, there are different ways to summarize the related disclosure risk. For example, one may compute the mean scores over all records, which is done in [15]. For our purposes, it seems more appropriate to focus on the measure discussed in the context of Reiter et al.'s approach in Section 2. Let j be an arbitrary record in $\mathcal{D}$. In Step (3), we additionally compute the attribution probability of all occurring target values y, that is

$$\mathrm{AP}_{s,j,y} := \mathrm{P}_s(y \mid K_{s,j}) = \frac{\sum_{i=1}^{n}[T_{s,i} = y \wedge K_{s,i} = K_{s,j}]}{\sum_{i=1}^{n}[K_{s,i} = K_{s,j}]}$$

for the synthetic datasets. $\mathrm{AP}_{o,j,y}$ is defined similarly. Analogous to Equation 2.1 in Section 2, we define

$$R_j := [\mathrm{CAP}_{s,j} = \mathrm{argmax}_y(\mathrm{AP}_{s,j,y})]$$

and compute $R = \sum_{i=1}^{m} R_j/m$, where $m = 1,473$ is the total number of records. Note that R corresponds to the accuracy of the related FR-NN classifier used by the attacker, and is therefore more interesting to us than the mean of the scores. However, we want to stress the fact that the following comparison between CAP and GCAP does not depend on focusing on accuracy, and we are able to draw similar conclusions by considering mean scores.

We now consider Table 1, which presents the scores for Scenario (1) and attribute key length three. The table summarizes the results of all possible keys amongst the variables in QI, that is, $C(7,3) = 35$ different situations. Each cell contains the average of the respective

0CAP, ICAP and GCAP scores

ConCep	0CAP	ICAP	GCAP
Original	54.9±7.8	54.9±7.8	54.9±7.8
DS 0	43.4±3.5	45.1±3.8	45.4±3.9
DS 0.1	40.6±5.4	42.4±4.3	43.2±3.9
DV	32.4±4.4	34.7±2.7	35.5±2.5
SP	46.0±2.3	47.4±3.2	47.6±3.3

Table 1: Scenario (1) / KL 3

ConCep	0CAP	ICAP	GCAP
Original	84.0±7.4	84.0±7.4	84.0±7.4
DS 0	22.6±6.1	45.0±2.3	46.1±2.3
DS 0.1	16.7±8.6	41.9±2.1	44.5±2.1
DV	14.5±4.9	35.5±1.5	35.6±0.7
SP	30.7±4.4	53.2±3.8	51.9±2.0

Table 2: Scenario (1) / KL 6

ConCep	0CAP	ICAP	GCAP
Original	64.3±3.1	64.3±3.1	64.3±3.1
DS 0	61.2±2.9	61.8±2.4	61.7±2.6
DS 0.1	60.9±3.8	61.5±3.0	61.4±3.2
DV	60.6±3.2	61.7±2.2	61.3±2.1
SP	62.4±2.9	62.7±2.7	62.6±2.7

Table 3: Scenario (2) / KL 2

ConCep	0CAP	ICAP	GCAP
Original	77.8±7.0	77.8±7.0	77.8±7.0
DS 0	52.5±6.3	62.0±3.3	60.9±2.6
DS 0.1	50.4±9.2	61.7±6.0	60.9±4.8
DV	44.5±8.9	61.2±3.8	59.6±2.3
SP	59.4±5.0	68.4±4.0	66.7±2.9

Table 4: Scenario (2) / KL 4

Non-Match	Table 1	Table 2	Table 3	Table 4
DS 0	53	725	14	236
DS 0.1	64	854	16	317
DV	103	865	27	398
SP	42	611	6	190

Table 5: Average number of samples ignored by ICAP

risks R over these 35 attribute keys, as well as the standard deviation. The table consists of three columns: 0CAP comprises the risk if non-matches are coded as 0, and ICAP shows the result for ignored non-matches. In the third column, we have the disclosure risk based on the GCAP measure. Table 2 presents the results for Scenario (1) and key length six, whereas the Tables 3 and 4 concern Scenario (2) with key lengths two and four.

We start by making general observations. GCAP results in a higher disclosure risk than 0CAP. Since $\text{GCAP}_{s,j} \geq \text{CAP}_{s,j}$ holds for all records j, this is no surprise. The difference is significant for the larger keys in the Tables 2 and 4, which is also plausible since larger keys lead to an increasing number of non-matches. We point out that, in all tables, the risks entailed by GCAP are close to the risks that result from ICAP. Note again that ICAP just ignores non-matches and is only taken over matches. The large differences between 0CAP and GCAP already indicated that the number of ignored samples is significant in the Tables 2 and 4. Table 5 shows the average number of samples ignored by ICAP in each situation. We recall that the original dataset consists of 1,473 samples. Since ICAP and GCAP coincide on matches, the differences between them result from the varying scores of GCAP on the ignored samples. Since these differences are small, this experiment corroborates our claim that GCAP is a useful extension of the CAP disclosure risk measure. Whenever $\text{CAP}_{s,j}$ is undefined, the computation of $\text{GCAP}_{s,j}$ allows the data provider to give an adequate estimate for the risk of the respective record. Furthermore, we see that ignoring a large amount of samples or assigning them risk 0 leads to an underestimation of the dataset's total risk.

We now focus on the differences between the synthesizers. For the DataSynthesizer with disabled Differential Privacy (DS 0) and the synthetic dataset generated by synthpop (SP), the risk entailed by GCAP is generally higher than for the Synthetic Data Vault (DV) and the data generated by the DataSynthesizer with Differential Privacy (DS 0.1). This result was to be expected, as the latter tools tend to produce a larger distortion of the data and, therefore, lead to lower disclosure risks. More interesting is the comparison between smaller and larger key sizes. Compared to Table 3, the risk entailed by GCAP decreases in Table 4 for all tools except for synthpop. The risk development for larger key sizes is interesting and unexpected, as the attacker's situation improves due to an increase in prior knowledge. For example, we observe a substantial disclosure risk increase on the original dataset. In Scenario (1), the GCAP score rises from 54.9 to 84.0, which is a consequence of the fact that the equivalence class for large key sizes often contains only one element, namely exactly the record of the respective victim individual. From the attacker's perspective, the intuition is that there might be better ways to exploit longer key sizes on synthetic datasets than using the classifier related to the GCAP measure. We therefore continued to study this problem by comparing the performance of several algorithms suitable for solving the attacker's classification problem related to the Scenarios (1) and (2). In Tables 6-9, we show the results for Naïve Bayes (NB), Support Vector Machine (SVM), K-NearestNeighbors (KNN), RandomForest (RF), Logistic Regression (LR) and the variant of the RadiusNearestNeighbor (FRNN) classifier described by Algorithm 3.1. As explained earlier, the accuracy scores of the latter coincide with the GCAP disclosure risk measure. We utilised the scikit-learn package[5] in Python and employed all algorithms with the standard parameter settings, to avoid unintended optimization.

[5] Version 0.20.3

Machine Learning Algorithms Accuracy for the Attacker's Classification Problem
Contraception Dataset

Table 6: Scenario (1) / KL 3

ConCep	NB	SVM	KNN	RF	LR	FR-NN	ENS
Original	47.4±2.8	49.9±3.6	48.5±7.0	54.4±7.5	45.7±2.6	54.9±7.8	51.7±5.1
DS 0	46.6±2.7	46.6±4.1	43.0±4.9	45.3±4.0	43.6±2.2	45.4±3.9	46.8±4.1
DS 0.1	44.7±2.7	45.1±2.8	40.7±4.1	42.4±4.2	42.7±0.6	43.2±3.9	45.2±3.0
DV	39.2±2.4	37.8±3.2	36.2±2.5	35.1±2.7	39.6±2.5	35.5±2.5	39.0±2.4
SP	47.0±2.5	48.7±3.7	44.6±3.7	47.4±3.1	45.7±2.6	47.6±3.3	48.8±3.8

Table 7: Scenario (1) / KL 6

ConCep	NB	SVM	KNN	RF	LR	FR-NN	ENS
Original	49.9±1.6	57.7±2.3	64.5±3.1	82.3±7.0	50.9±1.8	84.0±7.4	66.8±4.3
DS 0	50.9±1.9	50.7±3.2	47.3±2.7	47.8±2.9	48.0±1.9	46.1±2.3	51.4±2.8
DS 0.1	47.4±2.3	48.2±1.6	43.5±1.2	43.5±2.7	43.4±0.5	44.5±2.1	47.9±2.3
DV	38.0±0.8	36.1±1.4	35.6±1.3	36.4±1.6	39.5±1.4	35.6±0.7	38.4±1.0
SP	49.4±1.3	54.2±2.3	51.4±1.7	52.2±3.0	50.2±1.6	51.9±2.0	55.0±2.7

Table 8: Scenario (2) / KL 2

ConCep	NB	SVM	KNN	RF	LR	FR-NN	ENS
Original	62.6±2.3	63.0±2.1	57.4±7.8	64.2±3.1	62.8±2.1	64.3±3.1	63.2±2.1
DS 0	62.7±2.1	62.8±2.1	53.7±9.1	61.4±2.9	62.8±2.1	61.7±2.6	62.8±2.1
DS 0.1	62.7±2.1	62.2±1.6	52.8±8.7	61.6±2.0	62.4±1.8	61.4±3.2	62.7±2.1
DV	61.6±0.7	61.9±1.2	54.1±10.0	60.8±1.8	61.6±0.7	61.3±2.1	61.7±0.8
SP	62.3±1.9	62.9±2.1	58.9±4.7	62.1±3.1	62.7±2.1	62.6±2.7	62.9±2.0

Table 9: Scenario (2) / KL 4

ConCep	NB	SVM	KNN	RF	LR	FR-NN	ENS
Original	64.2±1.7	65.0±1.5	69.4±3.5	77.0±6.6	64.4±1.6	77.8±7.0	67.0±1.9
DS 0	64.1±1.9	64.3±1.7	60.8±3.0	59.9±3.8	64.6±1.7	60.9±2.6	64.4±1.8
DS 0.1	63.4±1.6	63.1±1.1	59.7±4.5	59.3±3.0	64.1±1.6	60.9±4.8	63.6±1.4
DV	62.4±0.7	62.0±0.6	59.6±1.6	58.8±2.4	62.2±0.6	59.6±2.3	62.3±0.7
SP	63.3±1.7	64.7±1.7	63.1±3.9	64.8±3.2	64.4±1.6	66.7±2.9	65.0±1.6

Notably, the performances of NB, SVM, KNN and LR do improve from Table 8 to Table 9. It appears that the adversary might benefit from considering not only FR-NN, but different classifiers to solve their problem. On the other hand, this raises a natural question: Which classifier should the attacker choose to maximize the accuracy of the prediction? The attacker is not able to perform an evaluation and then choose the algorithm that yields the highest accuracy scores. We therefore considered the using an ensemble classifiers[6], denoted as 'ENS' in the last column of the tables. The idea is that the attacker applies each of the six machine learning algorithms to their problem and then picks a prediction by implementing a majority vote on the results of the classifiers. Indeed, we observe that ENS generally scores above average. On the synthetic datasets of Scenario (1), ENS even exceeds all other classifiers in five out of eight cases.

The results indicate that, in addition to GCAP, the accuracy score of the ensemble classifier is also worth to be considered as possible disclosure risk measure by the data provider. In Scenario (2), the accuracy of the ensemble on the synthetic data is relatively close to the accuracy on the original data. Clearly, the performance on the real data is an important reference point, as it usually constitutes an upper bound for the performance on the synthetic data. For the evaluation of the utility of the ensemble for the attacker, we should

[6]https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.VotingClassifier.html

Figure 2: ENS scores in Table 8

also consider lower bounds in terms of the accuracy of dummy classifiers. A first baseline is given by generating predictions uniformly at random. If we suppose that the attacker already uses the synthetic dataset, the predictions can be generated based on the target attribute's distribution. For example, we may consider a dummy classifier that always predicts the most frequent value of the target attribute in the synthetic data (sometimes called the zero-rule classifier).

All four synthetic data generators preserve the distribution of attributes to some extent. Therefore, the most frequent value of the target attribute is the same for all datasets, which leads to the same constant prediction and, therefore, constant accuracy scores of the dummy classifier. For Scenario (1), this score is 42.7%; for Scenario (2), it is 61.0%. One might come to the conclusion that the accuracy scores of the ensemble, and hence the disclosure risk is still "small enough". On the synthetic datasets, 66% is never exceeded. However, to evaluate the general usefulness of data synthesis as privacy-preserving method, we have to consider not the absolute risk, but the decrease of disclosure risk relative to the original data. In this sense, the ensemble scores of *DS 0* and *SP* in the Tables 6 to 9 exceed the respective dummy classifier baselines by a substantial margin, which may become more obvious by taking a look at the scores on a number line in Figure 2.

On the other hand, all synthetic datasets prevent the attacker from exploiting larger attribute key sizes for re-identification, which is the most important reason for the high accuracy of FR-NN on the real data in Tables 7 and 9. Furthermore, the DataSynthesizer can be used with Differential Privacy to lower the disclosure risk, although the results for varying values of ε are rather unstable. The synthpop package also comes with many possibilities for achieving more privacy, such as removing replicated statistical uniques from the generated dataset. All these options, however, will affect the quality and the utility of the synthetic data, which should also be considered for assessing the results of the Synthetic Data Vault. The relation between the utility and the privacy of synthetic data is best described as trade-off.

It has to be stressed that further experiments on other datasets are necessary to establish more empirical evidence. We therefore complemented our detailed experiment on the Contraceptive Method Choice dataset by considering two attacker scenarios for the Fertility dataset [7]. This dataset consists of 100 records of volunteers that provided semen samples. In ten attributes, it comprises a variety of sensitive health information, such as whether the patient had child diseases, accidents, serious trauma, or surgeries. Further features are the frequency of alcohol consumption, smoking habits and, of course, the diagnosis of the semen sample. Since only few variables seemed to be adequate candidates for the set of quasi-identifiers, we focused on the following two scenarios:

(1) QI = {'age', 'alcohol', 'smoking habit'}
Target = 'accident'
(2) QI = {'age', 'alcohol', 'smoking habit'}
Target = 'surgery'

For both situations, we considered the average of the three attribute keys of length 2. Knowing only two of the three attributes in QI, the goal of the attacker is to infer whether their victim had an accident or surgery in the past. Tables 10 and 11 show the results. The dummy classifier baseline for the target 'accident' in Scenario (1) is 56%. Again, the ensemble exceeds this value substantially on *DS 0* and *SP*, as the performance of *DS 0* is actually close to the original data. For Scenario (2), the dummy baseline is 49% for *SP* and 51% for *DS 0*, *DS 0.1* and *DV*. We may draw similar conclusions, although this is the first situation in which not only the use of *DS* and *SP*, but also of *DS 0.1* and *DV* may lead to privacy breaches and considerable disclosure risk for certain records. In Figure 3, we consider the scores of ENS in Table 11 on a number line.

5 CONCLUSION AND FUTURE WORK

In this paper, we considered the problem of establishing a baseline for attribute disclosure risk on synthetic data. Given some prior knowledge in form of the values of several key attributes of a record of the original dataset and at least one synthesized dataset, what may the attacker infer about the record's entry for some sensitive target attribute? First of all, they may employ a *zero rule* classifier, which considers the distribution of the target attribute in the synthetic data and forms the prediction by choosing the most prominent entry. This straight-forward approach establishes a first baseline, but is superseded by other methods. We discussed Correct Attribution Probability, a recently published risk measure based on a matching mechanism, and generalized it to the GCAP measure, which also handles non-matches. In the evaluation, we saw that our approach improves the estimation of the disclosure risk, since it better reflects the ability of the adversary. Additional refinement of the accuracy scores is achieved by implementing several machine learning classifiers and employing an ensemble classifier, applying a majority vote on the obtained predictions of several individual classifier. We conducted our experiment by averaging over all possible attribute keys of certain length for a predefined set of quasi identifier variables, to provide an estimation of the average attack risks on all scenarios.

In Section 4, we saw that some of the evaluated synthetic datasets revealed sensitive information about the individuals in the original data. This can be prevented by using the disclosure control measures available to the user of the discussed tools. The influence on the quality and utility of the resulting synthetic data is certainly interesting and worth to be subject of further investigation. However, we point out that there are conceptual limits to the pursuit of keeping data utility and simultaneously decreasing disclosure risk. In the long-key scenario of Table 7, the Synthetic

[7] https://archive.ics.uci.edu/ml/datasets/Fertility

Machine Learning Algorithms Accuracy for the Attacker's Classification Problem
Fertility Dataset

Table 10: Scenario (1)

Fertility	NB	SVM	KNN	RF	LR	IRNN	ENS
Original	60.7±1.9	62.7±2.9	69.0±1.6	75.7±5.0	61.0±2.2	77.0±5.9	69.0±2.2
DS 0	61.3±2.1	60.3±3.1	58.0±2.9	66.0±3.6	63.7±1.7	67.0±0.0	68.7±2.1
DS 0.1	57.7±2.1	61.7±3.4	58.3±1.2	50.3±5.2	56.3±2.5	53.7±8.3	58.7±3.3
DV	56.7±3.3	56.7±0.5	51.7±5.4	54.0±4.2	59.3±2.6	50.7±7.1	55.3±3.3
SP	62.0±3.3	61.7±1.2	65.0±1.4	63.3±6.1	62.7±3.3	63.7±6.6	63.3±3.9

Table 11: Scenario (2)

Fertility	NB	SVM	KNN	RF	LR	IRNN	ENS
Original	60.3±5.2	60.3±3.3	65.0±7.1	70.3±8.1	57.7±5.4	71.7±9.0	64.7±4.0
DS 0	58.0±5.0	59.0±5.7	57.0±5.1	61.7±7.9	59.3±5.9	63.0±7.5	59.3±5.9
DS 0.1	57.7±8.3	59.0±5.7	57.3±6.2	56.3±5.4	59.0±7.9	56.7±6.1	58.0±6.2
DV	56.3±3.9	57.7±4.0	55.7±0.9	57.3±4.5	60.3±3.3	52.7±1.2	56.7±5.6
SP	55.3±2.4	56.3±2.9	60.3±1.7	60.7±3.9	58.0±3.6	62.7±3.3	60.7±2.9

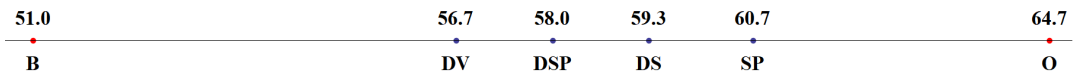


Figure 3: ENS scores in Table 11

Data Vault decreased the initially considerable disclosure risk of the original dataset down to the dummy classifier baseline. Obviously, this was not possible without also decreasing the utility of the synthetic dataset for training machine learning classifiers to predict the choice of contraceptive methods. Note that we just described one fact from two different perspectives. On an abstract level, the same property of the dataset has been altered by the synthesizer. This strong conflict between utility and disclosure prevention occurs whenever the target attribute in the applied classification task is a sensitive attribute. If the sensitive attribute is among the predictors, the problem is less drastic. In future work, we will therefore study the optimization problem of keeping data utility high and decreasing disclosure risk of sensitive predictor variables. Besides the mentioned experiments on other datasets, our future research will also concern the attacker's possibilities to make better use of prior knowledge and larger attribute keys. Finally, GCAP and all other concepts in this paper are only considered for categorical attributes. A generalization to continuous variables appears feasible.

ACKNOWLEDGMENTS

This work was partially funded by the KIRAS program (No 860663) of the Austrian Research Promotion Agency (FFG), the BRIDGE 1 programme (project WellFort, No 871267) of the Austrian Research Promotion Agency (FFG), and the EU Horizon 2020 research and innovation programme under grant agreement No 732907 (Project "MyHealthMyData"). The competence center SBA Research (SBA-K1) is funded within the framework of COMET — Competence Centers for Excellent Technologies by BMVIT, BMDW, and the federal state of Vienna, managed by the FFG.

REFERENCES

[1] Cynthia Dwork. 2006. Differential Privacy. In *Automata, Languages and Programming (Lecture Notes in Computer Science)*, Michele Bugliesi, Bart Preneel, Vladimiro Sassone, and Ingo Wegener (Eds.), Vol. 4052. Springer Berlin Heidelberg, Berlin, Heidelberg, 1–12. https://doi.org/10.1007/11787006_1

[2] Mark Elliot. 2014. *Final Report on the Disclosure Risk Associated with the Synthetic Data Produced by the SYLLS Team.* Technical Report. University of Manchester. http://hummedia.manchester.ac.uk/institutes/cmist/archive-publications/reports

[3] Benjamin C. M. Fung, Ke Wang, Rui Chen, and Philip S. Yu. 2010. Privacy-preserving data publishing: A survey of recent developments. *Comput. Surveys* 42, 4 (June 2010), 1–53. https://doi.org/10.1145/1749603.1749605

[4] Markus Hittmeir, Andreas Ekelhart, and Rudolf Mayer. 2019. On the Utility of Synthetic Data: An Empirical Evaluation on Machine Learning Tasks. In *Proceedings of the 14th International Conference on Availability, Reliability and Security - ARES '19.* ACM Press, Canterbury, CA, United Kingdom, 1–6. https://doi.org/10.1145/3339252.3339281

[5] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. 2006. L-diversity: privacy beyond k-anonymity. In *22nd International Conference on Data Engineering (ICDE'06).* 24–24. https://doi.org/10.1109/ICDE.2006.1 ISSN: 2375-026X.

[6] Beata Nowok, Gillian M. Raab, and Chris Dibben. 2016. synthpop: Bespoke Creation of Synthetic Data in R. *Journal of Statistical Software* 74, 11 (Oct. 2016). https://doi.org/10.18637/jss.v074.i11

[7] Neha Patki, Roy Wedge, and Kalyan Veeramachaneni. 2016. The Synthetic Data Vault. In *IEEE International Conference on Data Science and Advanced Analytics (DSAA).* IEEE, Montreal, QC, Canada, 399–410. https://doi.org/10.1109/DSAA.2016.49

[8] Haoyue Ping, Julia Stoyanovich, and Bill Howe. 2017. DataSynthesizer: Privacy-Preserving Synthetic Datasets. In *Proceedings of the 29th International Conference on Scientific and Statistical Database Management (SSDBM).* ACM Press, Chicago, IL, USA. https://doi.org/10.1145/3085504.3091117

[9] Jerome P. Reiter and Robin Mitra. 2009. Estimating Risks of Identification Disclosure in Partially Synthetic Data. *Journal of Privacy and Confidentiality* 1, 1 (April 2009). https://doi.org/10.29012/jpc.v1i1.567

[10] Jerome P. Reiter, Quanli Wang, and Biyuan Zhang. 2014. Bayesian Estimation of Disclosure Risks for Multiply Imputed, Synthetic Data. *Journal of Privacy and Confidentiality* 6, 1 (June 2014). https://doi.org/10.29012/jpc.v6i1.635

[11] Donald B. Rubin (Ed.). 1987. *Multiple Imputation for Nonresponse in Surveys*. John Wiley & Sons, Inc., Hoboken, NJ, USA. https://doi.org/10.1002/9780470316696

[12] H Surendra and H. S. Mohan. 2017. A Review Of Synthetic Data Generation Methods For Privacy Preserving Data Publishing. *International Journal of Scientific & Technology Research* 6, 3 (March 2017), 95–101.

[13] Latanya Sweeney. 2002. Achieving K-anonymity Privacy Protection Using Generalization and Suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10, 5 (Oct. 2002), 571–588. https://doi.org/10.1142/S021848850200165X

[14] Latanya Sweeney. 2002. K-anonymity: A Model for Protecting Privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10, 5 (Oct. 2002), 557–570. https://doi.org/10.1142/S0218488502001648

[15] Jennifer Taub, Mark Elliot, Maria Pampaka, and Duncan Smith. 2018. Differential Correct Attribution Probability for Synthetic Data: An Exploration. In *Privacy in Statistical Databases (Lecture Notes in Computer Science)*, Josep Domingo-Ferrer and Francisco Montes (Eds.). Springer International Publishing, Valencia, Spain, 122–137. https://doi.org/10.1007/978-3-319-99771-1_9