Privacy Evaluation in Tabular Synthetic Data: Current Approaches and Future Directions

Abstract

This paper examines the present methods for quantifying the level of privacy protection offered by tabular synthetic data (SD). Currently, there is no standardized approach for measuring the degree of privacy protection these datasets offer. This discussion contributes to the development of SD privacy standards, encourages interdisciplinary discourse, and aids SD researchers in making well-informed choices concerning modeling and assessment.

1 Introduction and Relation to Prior Research

Synthetic data (SD) has emerged as a powerful tool for enhancing privacy, preserving the analytic utility of data while decoupling it from real individuals. However, the wide variety of SD generation approaches makes the degree of privacy protection they offer difficult to assess. Therefore, this paper outlines the typical technical assessment frameworks for individual privacy in SD sets. This increases interdisciplinary awareness of privacy in SD and helps SD researchers make informed modeling and assessment choices.

While several surveys mention privacy as a use case for SD, they do not cover its assessment in a detailed way. In addition, reviews of privacy in AI fail to mention SD, and surveys, reviews, and experimental comparisons of SD techniques often do not focus on privacy metrics. Furthermore, legal analyses of SD are scarce and do not address quantitative methods for privacy assessment on a case-by-case basis.

2 Definitions and Notation

To the best of our knowledge, there is currently no widely accepted definition of SD. We present Definition 2.1, which is consistent with the approach by Jordon et al.

Definition 2.1. (Synthetic data) Synthetic data (SD) are data generated through a purpose-built mathematical model or algorithm (the "generator"), intended to solve a set of data science tasks.

We let D denote a database describing data subjects with attributes $A(D)$. Rows $d \in D$ are $|A(D)|$-tuples, with a value $v(d, a)$ for each attribute $a \in A(D)$. An attribute $a \in A(D)$ is categorical if its domain is finite and numerical if its domain is a subset of $\mathbb{R}$. We use the terms row and record interchangeably. We denote by G a generator, and $\hat{D} \sim G(D)$ to represent a synthetic dataset $\hat{D}$ obtained from generator G trained on D. Seed-based generators are a specific type of generators that produce a unique synthetic record denoted by $G(d)$ for every real record d. This is different from most models (e.g., GANs, VAEs) which probabilistically represent overall dataset properties and produce synthetic data by sampling from the obtained distribution.

.

3 Synthetic Data Privacy Risks

Three significant risks identified in prior works serve as a basis for a proper anonymization. These are: singling out, linkability, and inference. Privacy risks in SD can occur due to various factors, which include:

- *Model and data properties:* Improperly trained generators may overfit, memorizing and reproducing training data rather than inferring them stochastically. Records that emerge in isolation with little variability in their attribute values are difficult to generalize. As such, datasets containing outliers or sparse data are more at risk of memorization than more homogeneous sets. Such datasets are also more susceptible to singling-out.
- *The approach to data synthesis:* Most generators create stochastic models of datasets, creating synthetic records via sampling. This detaches real data subjects from synthetic records. However, some methods create a single synthetic record for each real record. This approach poses greater risk as it retains the link between a subject and its data.
- *Mode collapse:* GANs can focus on the minimal information necessary to deceive the discriminator, failing to capture the nuances and variations of the real data. In such cases, the SD resembles a small selection of real data subjects well, but not the entire population. This causes data clutter around specific real records, leaking their information.
- *The threat model:* A threat model describes the information an adversary leverages besides the SD. This can range from no access to the generator, to full knowledge including model parameters. Threat models also include scenarios where an adversary uses auxiliary information and can be:
  - No box: the adversary only has access to the SD.
  - Black box: the adversary also has limited generator access (no access to the model class or parameters, but access to the model2019s input-output relation).
  - White box: the adversary has full generator access (model class and parameters).
  - Uncertain box: the adversary has stochastic model knowledge (model class and knowledge that parameters come from a given probability distribution).
  - Any of the aforementioned, along with auxiliary information, which is formalized in the definition of auxiliary information in Definition 3.1.

Definition 3.1. Let D be a dataset with attributes $A(D)$. An adversary has auxiliary information if they know the values of a subset A' of attributes of some subset D' of records.

4 Mathematical Privacy Properties

4.1 Differential Privacy

Differential privacy (DP) is a property of information-releasing systems where data is not released directly, but via a processed version. The system is considered DP if the released information does not significantly change when one record is removed from the dataset.

Definition 4.1. (Differential Privacy) A randomized algorithm M is (ϵ, δ)-differentially private ((ϵ, δ)-DP) if, for all $S \subseteq A(P)$:

$$P[M(D) \in S] \leq e^{\epsilon} \cdot P[M(D') \in S] + \delta,$$

for all databases D, D' such that $\exists d \in D : D' = D \setminus \{d\}$. Generators are information-releasing systems and can therefore be DP. Suppose there are two real datasets, D and D', with $D' = D \setminus \{d\}$. A generator G is considered DP if a data controller with access to $\hat{D} \sim G$ cannot infer if G was trained on D or D'. Approaches to train generators with built-in mechanisms to guarantee DP can be found in the literature. In this context, DP is a property of generators, not of the synthetic data they produce.

4.2 k-Anonymity

Privacy risks persist, even if identifying attributes are removed. Combinations of attribute values may still be used to single out an individual. The notion of k-anonymity was introduced to address these

risks. A dataset is k-anonymous if at least k individuals share each combination of attribute values. Further restrictions such as l-diversity, t-closeness, and (α, k)-anonymity have been introduced to offer additional protection.

Synthetic data based on autoregressive models can implement k-anonymity directly into the generation process. For example, pruning in decision trees can guarantee that each combination of attribute values is sampled at least k times in mathematical expectation. Unlike DP, k-anonymity is a property of synthetic datasets, not the algorithms producing them.

4.3 Plausible Deniability

A degree of plausible deniability is inherent in synthetic datasets, as their records do not pertain to real data subjects. Two approaches have emerged to formalize this notion, with one most relevant to seed-based synthetic data.

Definition 4.2. (Plausible deniability) Let D be a dataset and G be a generator that converts any record $d \in D$ into a corresponding synthetic record $\hat{d} = G(d)$. For any dataset D where $|D| > k$, and any record $\hat{d}$ such that $\hat{d} = G(d_1)$ for $d_1 \in D$, we say that $\hat{d}$ is releasable with (k, γ)-plausible deniability if there exist at least $k - 1$ distinct records $d_2, ..., d_k \in D \setminus \{d_1\}$ such that for all $i, j \in \{1, 2, ..., k\}$:

$$P[d = G(d_i)] \approx_\gamma P[d = G(d_j)]$$

In other words, a generator producing synthetic records from a seed has PD if, for each synthetic record produced from a particular seed, k other seeds could have resulted in roughly the same (quantified through γ) synthetic record. Like DP, and unlike k-anonymity, PD is a property of (seed-based) generators, though it is related to both.

5 Statistical Privacy Indicators

5.1 Identical Records, Distances, and Nearest Neighbors

Most indicators quantify the frequency of synthetic records being identical or suspiciously similar to real records. Unlike DP and PD, these indicators measure properties of synthetic datasets, not their generators. The proportion of synthetic records that match real records is called the identical match share (IMS). The IMS has been generalized to similarity metrics, and further to Nearest neighbor (NN)-based methods. These can be classified based on the following properties, summarized in Table 3 of Appendix C:

- *Similarity metrics.* Table 2 of Appendix C contains an overview of commonly invoked measures.
- *Metric evaluation.* Because structured datasets can have a mix of different datatypes, metric evaluation is complex. Several approaches exist, such as binning numeric attributes; combining multiple metrics; ignoring specific attributes; or evaluating distances in embedding spaces.
- *Evaluated distances.* For a given synthetic record $\hat{d} \in \hat{D}$, we can find its closest real record $d \in D$. The distance between these records is the synthetic to real distance (SRD) of $\hat{d}$, and is denoted as $SRD(\hat{d})$:

  $$SRD(\hat{d}) := \min_{d \in D} \text{Dist}(\hat{d}, d) \qquad \forall \hat{d} \in \hat{D}.$$

  Similarly, the smallest synthetic-to-synthetic (SSD), real-to-synthetic (RSD), and real-to-real distance (RRD) can be defined.
- *Use of holdout sets.* To compute the RRD, the real data D can be partitioned into two subsets D_1 and D_2. For a real record $d_1 \in D_1$, the RRD is the smallest distance to any record $d_2 \in D_2$:

  $$RRD(d_1) := \min_{d_2 \in D_2} \text{Dist}(d_1, d_2) \qquad \forall d_1 \in D_1.$$

  This provides a baseline for SD comparison.

- *Statistics.* The distance to the closest record (DCR) compares the SRD and RRD distributions. Statistical properties are expressed through the proportions of "suspiciously close" synthetic records. Measures used for this include medians, means, and standard deviations. Small percentiles are also often invoked when analyzing the distance distribution.

5.2 Other Statistical Indicators

The targeted correct attribution probability (TCAP) is an indicator of parameter inference attack success rates. It measures how often synthetic parameter values correspond to real values in l-diverse equivalence classes. Furthermore, there are several probabilistic techniques to quantify the risks by using real hold-out sets as baselines. Maximum mean discrepancy (MMD) can be also used as a privacy metric to test if the generator overfits.

6 Computer Scientific Experimental Privacy Assessment

Computer-scientific privacy assessment involves performing privacy attacks using synthetic data. The effectiveness of these attacks is used to measure the degree of protection SD provides. Attack frameworks, as classified in Table 4 of Appendix D, are based on threat models and the following factors:

- *Attack Frameworks.* These include Vulnerable Record Discovery (VRD), which identifies synthetic records that are the result of overfitting generators. Other frameworks include Model inversion, membership inference attacks (MIAs), and shadow modeling, which can all compromise confidentiality.
- *Attack Mechanisms.* Nearest Neighbors (NN) is one such attack mechanism, where an adversary infers missing attribute values based on its k synthetic nearest neighbors. Machine learning (ML) techniques are another approach, where classifiers are trained to re-identify real data subjects. Additionally, information theory (IT) measures, such as Shannon entropy and mutual information, are sometimes used to identify records that may be more likely to be memorized by the generator.
- *Baselines and Effectiveness Estimation.* The efficacy of a model can be measured in a few different ways. Absolute metrics include the probability with which records can be singled out, and the proportion of real records that can be re-identified. A random baseline approach uses random guessing to determine how effective an attack is. In a control baseline, the real data is split into a training set and a control set. A model is trained on the training set, and then the estimated success rate of attacks is compared on the training and control data sets. Another approach involves the deliberate insertion of secrets in training data or in the SD after generation.

6.1 Relation to WP29 Attack Types

- *Singling out.* VRD attacks directly implement singling-out attacks, identifying outlier SD records. MIAs can also model singling out, where an adversary quantifies the likelihood of a unique real record's attribute combination.
- *Linkage.* NN-based attacks usually require auxiliary information and can be interpreted as linkage attacks. Anonymeter and information theory based VRD are the only methods that explicitly model linkage attacks.
- *Inference.* NN-based attacks and MIA can be seen as inference attacks.

7 Discussion

7.1 The Assessment Frameworks

Mathematical privacy properties, such as differential privacy (DP), do not offer a clear choice of the required parameters (ϵ, δ). Large parameter values offer weak privacy guarantees, and a given ϵ can result in different degrees of protection depending on the application. DP may still be vulnerable to linkage and inference attacks, giving a false sense of security, and is a property of generators and

not their synthetic data. The difficulty with k-anonymity is that implementing it causes considerable information loss and is an NP-hard problem. Furthermore, k-anonymity was shown to offer sufficient protection only when the utility of the data is completely removed. In addition, k-anonymity is a property of synthetic data, and not the methods to produce them. Plausible deniability (PD) is only applicable to seed-based methods. It shares properties with both DP and k-anonymity, making a record protected if it can be confused with other records.

Statistical privacy indicators are difficult to interpret, with many options and decision points, such as the choice of similarity metric. Statistical indicators measure properties of the synthetic data, and not their generators.

Computer-scientific experiments allow for flexible modeling using various threat models, and can include properties of both synthetic data and their generators. However, they require more data and computation than mathematical properties.

7.2 Relation to Synthetic Data Risks

All assessment frameworks address the issue of generator memorization. Mathematical properties focus on the uniqueness of records. DP measures the impact of individual training records, with outliers having large impacts, and both k-anonymity and PD focus on limiting the uniqueness of records. Distance-based indicators are sensitive to outliers, because synthetic neighbors of outliers have small SRDs, while the RRD of corresponding real outliers is large. Furthermore, some methods explicitly search for outliers.

There are currently no studies that assess whether seed-based generators inherently pose greater risks than other generators.

7.3 Suggestions for Future Research

For the future research directions we identify are:

- *Standardizing privacy assessment:* More interdisciplinary research is required to develop an inclusive understanding of synthetic data. Standards should be developed for research findings to be more easily interpreted, and there should be a consensus formed over whether privacy is a property of synthetic datasets, the generators, or both.
- *Synergies between assessments:* A comparison between mathematical, statistical, and empirical approaches would be useful to evaluate their consistency, and to identify their individual merits and weaknesses. Experiments should use open-source generators and publicly available datasets. It would also be useful to include information regarding the used metrics, and the use of a holdout set, and the statistical interpretation of the results.
- *Outlier protection:* Future research should investigate methods for outlier protection through binning and aggregating attributes or using innovative techniques. It would also be beneficial to see how outlier detection can be used to guide vulnerable record discovery.
- *Incorporating privacy into generators:* While DP is used in some generators, the same is not true for all privacy metrics and empirical privacy methods. Future research should focus on incorporating these, by integrating metrics in loss functions, or by combinatorial optimization.
- *Assessment for advanced data formats:* More work is needed to assess privacy in relational datasets that have information contained in multiple, interconnected tables. In particular, profiling attacks, which re-identify subjects based on behavioral patterns, may play a key role in the assessment of relational databases.
- *Distribution-level confidentiality:* There is a need for frameworks that assess the confidentiality of overall dataset properties.

A A Proof of Theorem 2.1

Proof. Let $x \in A_i$. Then, $\sigma_i(x) = 0$, and for all $b \in O$ where $b_i = 0$, $w_b(x) = 0$. Thus,

$$F(x) = \sum_{b \in O, b_i = 1} w_b(x) G_b(x)$$

If $b_i = 1$, then $G_b(x) \in B_i$, and therefore $F(x)$ is also in B_i due to the convexity of B_i.

B B Example on Synthetic Datasets

Figure 2 depicts an example of applying our safe predictor to a notional regression problem with 1-D input and outputs, and one input-output constraint. The unconstrained network has a single hidden layer of dimension 10 with ReLU activations, followed by a fully connected layer. The safe predictor shares this structure with constrained predictors, G_0 and G_1, but each predictor has its own fully connected layer. The training uses a sampled subset of points from the input space and the learned predictors are shown for the continuous input space.

Figure 3 shows an example of applying the safe predictor to a notional regression problem with a 2-D input and 1-D output and two overlapping constraints. The unconstrained network has two hidden layers of dimension 20 with ReLU activations, followed by a fully connected layer. The constrained predictors G_{00}, G_{10}, G_{01} and G_{11} share the hidden layers and have an additional hidden layer of size 20 with ReLU followed by a fully connected layer. Again, training uses a sampled subset of points from the input space and the learned predictors are shown for the continuous input space.

C C Details of VerticalCAS Experiment

C.1 Safeability Constraints

The "safeability" property from previous work can be encoded into a set of input-output constraints. The "safeable region" for a given advisory is the set of input space locations where that advisory can be chosen, for which future advisories exist that will prevent a NMAC. If no future advisories exist, the advisory is "unsafeable" and the corresponding input region is the "unsafeable region". Figure 5 shows an example of these regions for the CL1500 advisory.

The constraints we enforce in our safe predictor are: $x \in A_{\text{unsafeable},i} \Rightarrow F_i(x) < \max_j F_j(x)$, $\forall i$. To make the output regions convex, we approximate by enforcing $F_i(x) = \min_j F_j(x)$, for all $x \in A_{\text{unsafeable},i}$.

C.2 Proximity Functions

We start by generating the unsafeable region bounds. Then, a distance function is computed between points in the input space ($v_O - v_I$, h, τ), and the unsafeable region for each advisory. These are not true distances, but are 0 if and only if the data point is within the unsafeable set. These are then used to produce proximity functions. Figure 5 shows examples of the unsafeable region, distance function, and proximity function for the CL1500 advisory.

C.3 Structure of Predictors

The compressed policy tables for ACAS Xu and VerticalCAS use neural networks with six hidden layers with a dimension of 45, and ReLU activation functions. We used the same architecture for the unconstrained network. For constrained predictors, we use a similar architecture, but share the first four layers for all predictors. This provides a common learned representation of the input space, while allowing each predictor to adapt to its constraints. Each constrained predictor has two additional hidden layers and their outputs are projected onto our convex approximation of the safe output region, using $G_b(x) = \min_j G_j(x) - \epsilon$. In our experiments, we used $\epsilon = 0.0001$.

With this construction, we needed 30 separate predictors to enforce the VerticalCAS safeability constraints. The number of nodes for the unconstrained and safe implementations were 270 and 2880, respectively. Our safe predictor is smaller than the original look-up tables by several orders of magnitude.

C.4 Parameter Optimization

We use PyTorch for defining our networks and performing parameter optimization. We optimize both the unconstrained network and our safe predictor using the asymmetric loss function, guiding the

network to select optimal advisories while accurately predicting scores from the look-up tables. Each dataset is split using an 80/20 train/test split with a random seed of 0. The optimizer is ADAM, with a learning rate of 0.0003, a batch size of 216, and training for 500 epochs.