

# TableGAN-MCA: Evaluating Membership Collisions of GAN-Synthesized Tabular Data Releasing

Aoting Hu  
Southeast University  
China

Renjie Xie  
Southeast University  
China

Zhigang Lu  
Macquarie University  
Australia

Aiqun Hu  
Southeast University  
China

Minhui Xue  
The University of Adelaide  
Australia

## ABSTRACT

Generative Adversarial Networks (GAN)-synthesized table publishing lets people privately learn insights without access to the private table. However, existing studies on Membership Inference (MI) Attacks show promising results on disclosing membership of training datasets of GAN-synthesized tables. Different from those works focusing on discovering membership of a given data point, in this paper, we propose a novel Membership Collision Attack against GANs (*TableGAN-MCA*), which allows an adversary given only synthetic entries randomly sampled from a black-box generator to recover partial GAN training data. Namely, a GAN-synthesized table immune to state-of-the-art MI attacks is vulnerable to the *TableGAN-MCA*. The success of *TableGAN-MCA* is boosted by an observation that GAN-synthesized tables potentially collide with the training data of the generator.

Our experimental evaluations on *TableGAN-MCA* have five main findings. First, *TableGAN-MCA* has a satisfying training data recovery rate on three commonly used real-world datasets against four generative models. Second, factors, including the size of GAN training data, GAN training epochs and the number of synthetic samples available to the adversary, are positively correlated to the success of *TableGAN-MCA*. Third, highly frequent data points have high risks of being recovered by *TableGAN-MCA*. Fourth, some unique data are exposed to unexpected high recovery risks in *TableGAN-MCA*, which may attribute to GAN's generalization. Fifth, as expected, differential privacy, without the consideration of the correlations between features, does not show commendable mitigation effect against the *TableGAN-MCA*. Finally, we propose two mitigation methods and show promising privacy and utility trade-offs when protecting against *TableGAN-MCA*.

## CCS CONCEPTS

• Security and privacy; • Computing methodologies → Machine learning;

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

CCS 2021, 14 - 21 November, 2021, Seoul, South Korea

© 2021 Association for Computing Machinery.

ACM ISBN 978-1-4503-XXXX-X/18/06...\$15.00

## KEYWORDS

Membership Privacy, Differential Privacy, Generative Adversarial Networks (GANs), Synthetic Data Releasing

### ACM Reference Format:

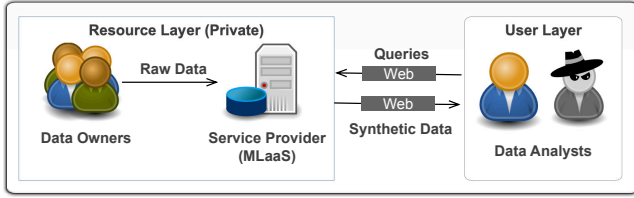
Aoting Hu, Renjie Xie, Zhigang Lu, Aiqun Hu, and Minhui Xue. 2021. TableGAN-MCA: Evaluating Membership Collisions of GAN-Synthesized Tabular Data Releasing. In *2021 ACM SIGSAC Conference on Computer and Communications Security (CCS '21)*, November 14–19, 2021, Virtual Event, South Korea. ACM, New York, NY, USA, 16 pages.

## 1 INTRODUCTION

Big data have emerged as valuable resources that allow companies, researchers and governments to enhance decision making, insight discovery and process optimization. However, sharing sensitive datasets without violating individual's privacy is a long-standing challenge. For example, in 2017, DeepMind was accused of an illegal acquisition of personal medical records of 1.6 million patients for developing a kidney injuries diagnosing application [34]. To analyze those sensitive data in a privacy-preserving manner, ideally, we need a trusted third party that collects and processes raw data, and then releases a sanitized version of data trading off privacy and utility through web queries (see the paradigm shown in Fig. 1).

However, state-of-the-art solutions for releasing the sanitized data achieving trade-offs between utility and privacy are vulnerable to privacy inference attacks. For example, de-identification (removing unique identifiers for all data entries) is susceptible to linkage attacks [32]. Anonymization [24, 29, 45] suffers from background information attacks. Other synthetic dataset publishing mechanisms, such as NetMechanism [4], Iterative Construction [16–19], are tailored for relatively small datasets [13]. More recently, Generative Networks, including Generative Adversarial Networks (GANs) [14] and Variational Autoencoders (VAEs) [23], produce synthetic data that achieve enhanced privacy and utility trade-offs. Such synthetic data conceal the detailed (privacy) of the raw data while keeping statistics similarity [35, 46]. Nevertheless, recent works [7, 20, 21, 35, 44] show the risk of membership disclosure (i.e., inferring whether a given data point belongs to the training dataset) against synthesized data by attacking generator APIs. They propose various Membership Inference Attacks (MIAs) against published generative models to disclose the membership information of training data.

To further explore the privacy disclosure risks of the GAN-synthesized tabular data, different from existing MIAs against generative models [7, 20, 21, 35, 44], we propose a novel attack model,



**Figure 1: The framework of private data publishing. Both the data owner and service provider who guard resources are trusted. The data analysts are legal customers as well as potential adversaries.**

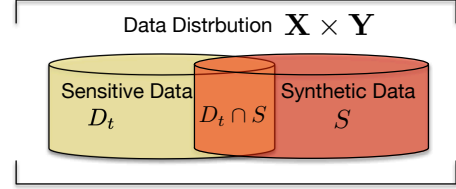
named Membership Collision Attack against GAN-synthesized Tables (*TableGAN-MCA*). Specifically, we reconstruct a proportion of actual training data from the published synthetic table with high confidence by inferring the membership collisions (substantiated in Section 3.1). Hence, *TableGAN-MCA* brings a novel privacy problem: training data exposure when analyzing published synthetic tabular data. In addition, *TableGAN-MCA* only queries a black-box generator (of the GAN) for synthetic data, which is similar to the most strict threat model introduced in the recent work - GAN-Leaks [7]. We conceptualize the differences among recent works in Table 1. **Motivation.** Our work is motivated by two observations in GAN-synthesized table (low-dimensional data) releasing.

- **Observation 1.** Generated synthetic tables overlap with GAN’s training data (as the intersection illustrated in Fig. 2). For instance, in the Adult dataset, a synthetic dataset collides with the GAN’s training dataset by 16.9% (5350 entries). Clearly, such an overlap brings severe privacy breaches if adversaries could locate the intersection. In the remainder of this paper, we call the overlap/intersection *membership collision*.
- **Observation 2.** In the GAN-synthesized tabular data, membership collisions and data frequency are positively correlated (substantiated in Fig. 3). However, it is rare to trigger sample collisions in high-dimensional data, such as image synthesis, due to the curse of dimensionality. Thus, the distribution of tabular data with relatively small dimension brings additional privacy risks than that of image synthesis.

To perform the proposed *TableGAN-MCA*, we leverage shadow models [43] to learn the patterns behind the collision (*Observation 1*) while taking the density of each synthetic data by counting its sample frequency in synthetic distribution (*Observation 2*) as additional feature when training the attack model. *TableGAN-MCA* shows promising results on commonly used real-world datasets, including Adult, Lawschool and Compas. For instance, ***TableGAN-MCA* recovers 36.1%, 12.7%, 36.5% of actual members released with the GAN-synthesized tabular data with approximately 80% confidence for Adult, Lawschool and Compas, respectively.** Our results show that a well-trained GAN, robust to the MIAs proposed in [7, 20, 35], is still vulnerable to *TableGAN-MCA*.

In summary, our main contributions are as follows:

- We propose a novel membership collision attack against GAN-synthesized tabular data publishing, named *TableGAN-MCA*, which can reinstate partial training data with high confidence.



**Figure 2: The training dataset  $D_t$  intersects the synthetic dataset  $S$  at  $D_t \cap S$ .**

**Table 1: Comparison with MIAs against GANs. (■: black-box access; −: insufficient information provided; ✓: require; ×: does not require)**

	Benchmark Datasets	■ Gen-erator	■ Dis-criminator	Extra Targets	Expose Trainset
LOGAN [20]	Image	✓	✓	✓	False
table-GAN [35]	Table	✓	✓	✓	False
MC [21]	Image	✓	×	✓	−
GAN-leaks [7]	Image/Table	✓	×	✓	False
<b>TableGAN-MCA</b>	Table	✓	×	×	True

*TableGAN-MCA* exploits the weaknesses of GAN synthesis observed on low-dimensional data, i.e., GAN-synthesized data collide with its training data, and members (in the colliding member set) occur more frequently than non-members.

- We extensively evaluate our proposed attacks on three commonly used real-world datasets, including Adult, Lawschool and Compas against four generative models, including TVAE [46], CTGAN [46], WGAN-GP [15] and WGAN-WC [2]. Furthermore, we explore the factors that may impact the attack effectiveness, such as the size of GAN training data, GAN training epochs, GAN training data frequencies and the number of synthetic samples available to the attacker.
- We discover that individuals in the training dataset have various risks of privacy leakage under *TableGAN-MCA*. Additionally, we show that GANs do not memorize those exposed data. Instead, when generalizing the distribution of the training data, GANs may increase or decrease the frequency of some individuals, and hence change their privacy risks.
- We examine the effect of differential privacy (DP) to mitigate *TableGAN-MCA*. Our empirical results show that differential private generative model training achieves sub-optimal trade-offs against *TableGAN-MCA*. It is mainly due to the fact that *TableGAN-MCA* relies more on the common pattern of a distribution (like attribute correlations) which is not the focus of DP. In addition to DP, we propose two mitigation methods, naive defense and improved defense, that mitigate the effect of *TableGAN-MCA*.

## 2 BACKGROUND OF GENERATIVE MODELS

Generative Adversarial Networks (GANs) [14] and its variants have made great achievements in generating high quality artificial data that mimic the real ones, by modeling the underlying data distribution. It is composed of two neural networks: a discriminator  $D$  and a generator  $G$ . It tries to minimize the distance between the real data distribution  $P_r$  and the generated (artificial) data distribution  $P_g$  by iteratively updating parameters of the networks.

**Table 2: Summary of notations.**

Symbol	Description	Symbol	Description
$D_t$	Private training dataset	$D_s$	Test dataset
$S$	Released synthetic dataset	$\tilde{S}$	Shadow dataset
$P_r$	Training data distribution	$G$	Generator oracle
$P_z$	Prior Gaussian distribution	$\mathbb{1}$	Indicator function
$P_g$	Generated data distribution	$\mathbf{x}$	A data point
$I$	colliding member set	$\mathcal{A}$	Adversary
$N_s$	Number of synthetic copies	$f(\cdot)$	Attack classifier

The Wasserstein GAN (WGAN) [2] applies Earth Mover (EM) distance under a K-Lipschitz constraint and achieves good performance in generating high fidelity samples. The loss function of the discriminator and the generator are as follows:

$$J^{(D)}(\theta^{(D)}, \theta^{(G)}) = -\frac{1}{2}\mathbb{E}_{\mathbf{x} \sim P_{\text{data}}} D(\mathbf{x}) + \frac{1}{2}\mathbb{E}_z D(G(z)), \quad (1)$$

$$J^{(G)} = -\frac{1}{2}\mathbb{E}_z D(G(z)). \quad (2)$$

In this work, we use its weight clipping version (WGAN-WC) [2], Gradient Penalty version (WGAN-GP) [15] and CTGAN (state-of-the-art) [46]. We also include TVAE from [46] for its comparable performance as CTGAN. Following [46], all three GANs uses recurrent networks in the generator. For categorical features, we use the gumble-softmax activation in the output of the generator. For numerical features, we use the sigmoid or the tanh activation in the output of the generator based on value range. The architecture and parameters of GANs are broken down in Appendix A.

### 3 PROBLEM FORMULATION

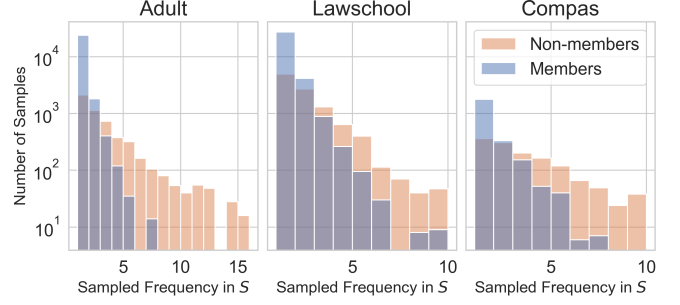
In this section, we formulate our membership collisions problem, followed by the description of the threat model according to adversary’s goals, capabilities and background knowledge. We introduce all the notations used throughout the paper in Table 2.

#### 3.1 Membership Collision Problem

We let  $D_t = \{\mathbf{x}\}$  be a training set sampled from an implicit data distribution  $P_r$ . Each private entry takes the form as  $\mathbf{x} = (x, y) \in \mathbf{X} \times \mathbf{Y}$ , where  $x$  represents the features and  $y$  represents the class label. A data release mechanism GAN trains on the training set  $D_t$  and outputs a well learned generator  $G$ . Generator  $G$  is a deterministic function that maps a prior distribution, i.e., Gaussian distribution  $P_z$ , to the generated distribution  $P_g$  that mimic real distribution  $P_r$ . Then, a synthetic dataset  $S \sim P_g$  is published and serves as a sanitized version of  $D_t$ . We formalize the **membership collisions** as : a published synthetic datasets  $S \sim P_g$  collide with its training set  $D_t \sim P_r$  and result in a colliding member set  $I = S \cap D_t$ . Notice that a data point  $\mathbf{x} \in I$  result in  $\mathbf{x} \in D_t$ . Similarly, a synthetic data point  $\mathbf{x} \notin I$  result in  $\mathbf{x} \notin D_t$ .

We aim to study how much an adversary  $\mathcal{A}$  increases its ability to assert whether a synthetic data point  $\mathbf{x} \sim S$  belongs to the colliding member set  $I$  by estimating the generated distribution  $P_g$  via the published synthetic dataset  $S$ . Formally,

*Definition 3.1 (Membership Collision Attack).* Given a synthetic dataset  $S$  produced by a generative model  $G(P_z, D_t)$  that contains a colliding member set  $I = S \cap D_t$  and an attack algorithm  $\mathcal{A}(\mathbf{x})$  that outputs 1 if it outputs the synthetic data  $\mathbf{x} \in I$ , we say the



**Figure 3: Comparisons of sample frequency between members and non-members. The x-axis represents all possibilities of  $\{\#x_i\}$  in published synthetic datasets and y-axis represents log of the number of eligible data points.**

generative model  $G$  is subject to membership collision inference attack if there exists an entry  $\mathbf{x} \in S$  such that

$$\Pr[\mathcal{A}(\mathbf{x}, P_g) = 1] - \Pr[\mathcal{A}(\mathbf{x}) = 1] > \alpha, \quad (3)$$

where  $\alpha$  is a non-negligible value.

In this work, we consider that the prior advantage of the attacker is random guess, that is,  $\Pr[\mathcal{A}(\mathbf{x}) = 1] = \Pr[\mathbf{x} \in I]$ . Thus, we evaluate the posterior advantage of the attacker thereafter.

Note that Def. 3.1 differs from the membership inference definition [43] by changing the goal of arbitrary membership inference with membership collision inference of synthetic data. The proposed *TableGAN-MIA* is an instance of MCA in GAN-synthesized table releasing.

#### 3.2 Threat Model

In the context of GAN-synthesized data sharing, adversaries are external parties that wish to learn the statistics of the sensitive dataset by querying data owners or curators. In existing MIAs against GANs [7, 20, 21, 35], the adversary’s knowledge is: (1) having only limited synthetic data, (2) accessing a black-box generator API (unlimited synthetic data), (3) accessing a black-box generator plus a discriminator oracle, (4) accessing a white-box GAN. Our study focus on the most strict attack model: (1) and (2) (which is similar to the threat model in MC [21] and “Full Black-box Generator” assumption in GAN-Leaks [7]). The attacker does not know the priori of the model’s structure, including meta-parameters, training data and any target data to infer membership. In *TableGAN-MCA*, the adversary’s goal is to recover the value of some members of the training set from the published synthetic datasets that may unintentionally contain colliding members. In this paper, we evaluate *TableGAN-MCA* under two threat models:

**Attack model (1):** accessible to limited synthetic data. We assume the adversary has one copy of synthetic dataset  $S$  following  $P_g$ , of size  $|S| = |D_t| = n$ .

**Attack model (2):** accessible to unlimited synthetic data. We assume the adversary has  $N_s$  ( $N_s$  is a positive integer) synthetic copies  $\{S_1, S_2, \dots, S_{N_s}\}$ , each of which has size  $|S_i| = |D_t| = n$ .

## 4 MEMBERSHIP RECOVERY FRAMEWORK AGAINST GAN-SYNTHEZIZED TABLES

In this section, we propose a membership indicator for inferring membership collisions from the statistics of the published table. Based on the membership indicator, we propose the *TableGAN-MCA* to recover the value of the training set of GAN-synthesized tables in the black-box setting.

### 4.1 Membership Indicator

The membership indicator is triggered by two observations. First, the released GAN-synthesized tables often overlap the training dataset of the GAN model. Second, such synthetic data points appearing frequently in the published GAN-synthesized data are more likely to be the colliding member of the training dataset. That is,  $\Pr[\mathbf{x} \in D_t | \mathbf{P}_g] \propto \Pr[\mathbf{x} | \mathbf{P}_g]$ . Fig. 3 depicts the observations from three datasets used in this paper, where we count the numbers of members and non-members, given numbers of appearance of the data points in the released synthetic tables. In Fig. 3 (left), approximately 96% of synthetic data with a sampled frequency of more than three are colliding members. Conversely, almost 91% unique synthetic data are non-colliding members in the Adult dataset. Thus, sample frequency is highly correlated with membership collisions and can be treated as an indicator to indicate membership. Formally, we estimated the membership indicator by the following equation.

$$\Pr[\mathbf{x}_i | \mathbf{P}_g] \approx \mathbb{E}_{\mathbf{x}_j \in S} \mathbb{1}(\mathbf{x}_i = \mathbf{x}_j) = \frac{1}{n} \sum_{j=1}^n \mathbb{1}(\mathbf{x}_i = \mathbf{x}_j), \quad (4)$$

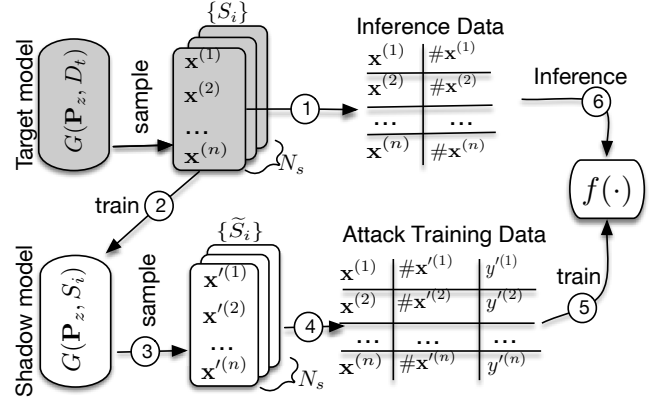
where an indicator function  $\mathbb{1}(\cdot)$  outputs 1 if its argument is true,  $S$  is the synthetic datasets available to the adversary following  $\mathbf{P}_g$ , of size  $|S| = n$ .

To date, the adversary can launch a data reconstruction attack by setting a threshold for the value of a membership collisions indicator of Eq. (4), similar to [7]. The adversary then claims that the synthetic data, having collisions indicators greater than a given threshold, are the recovered data. However, choosing an optimal threshold is a non-trivial task for an adversary without background knowledge about training data except the published synthetic data. To deal with it, we additionally leverage shadow model techniques [43] to enhance the knowledge of adversaries to construct a robust *TableGAN-MCA* framework.

### 4.2 TableGAN-MCA

In a nutshell, *TableGAN-MCA* combines the membership collisions indicator and the shadow models [43] to train an attack model to learn the relation between membership collisions (labels) and indicator values (features) in released GAN-synthesized tables. Fig. 4 depicts the framework of *TableGAN-MCA* and Alg. 1 shows the detailed implementation. Each step in Alg. 1 corresponds to the step index in Fig. 4. In summary, steps 2, 3, 4 and 5 train an attack classifier by giving synthetic data. Steps 1 and 6 infer membership collisions to recover training data.

In Steps 1 and 4,  $\{\#\mathbf{x}\}$  represents estimated sample frequency following from Eq. 4. They are concatenated (" $\bowtie$ ") to  $S_i$  (Step 1) and  $\tilde{S}_i$  (Step 4) as an extra feature.



**Figure 4: The overview of the procedures of *TableGAN-MCA* against the black-box generator in data synthesis.**

---

#### ALGORITHM 1: TableGAN-MCA.

---

**Input:**  $\{S_1, S_2, \dots, S_{N_s}\}$ : Released synthetic datasets;  $|D_t|$ : Size of the training dataset;

**Output:**  $R$ : Recovered data from  $D_t$

**while**  $i : 1 \rightarrow N_s$  **do**

**Step 1:**

    Frequency  $\{\#\mathbf{x}_i\} \leftarrow$  Estimate frequency for each  $\mathbf{x}_i \in S_i$  by Eq. (4);

$S_i \leftarrow S_i \bowtie \{\#\mathbf{x}_i\}$ ;

**Step 2:** Shadow GAN generator  $\tilde{G}_i \leftarrow$  Train on  $S_i$ ;

**Step 3:** Shadow set  $\tilde{S}_i \leftarrow$  Sample from  $\tilde{G}_i$ ,  $|\tilde{S}_i| = N_s \times |D_t|$ .

**Step 4:**

    Frequency  $\{\#\mathbf{x}'_i\} \leftarrow$  Count the frequency for each  $\mathbf{x}'_i \in \tilde{S}_i$  by Eq. (4);

$\tilde{S}_i \leftarrow \tilde{S}_i \bowtie \{\#\mathbf{x}'_i\}$ ;

    Ground truth label  $y'_i \leftarrow \mathbb{1}(\mathbf{x}'_i \in \tilde{I}_i)$ ;

**end**

**Step 5:** TableGAN-MCA attack model  $f(\cdot) \leftarrow$  Train on  $\|\_{i=1}^{N_s} \tilde{S}_i$  with member/non-member labels  $\|\_{i=1}^{N_s} \{y'_i\}$ , where  $\|\_{i=1}^{N_s} \tilde{S}_i = \tilde{S}_1 \parallel \dots \parallel \tilde{S}_{N_s}$ ;  $\|\_{i=1}^{N_s} \{y'_i\} = \{y'_1\} \parallel \dots \parallel \{y'_{N_s}\}$ ;

**Step 6:**  $R_A \leftarrow f(\{S_i\})$ ;

**return**  $R_A$

---

In Step 4, a label function is required to claim membership collisions in shadow datasets. For a shadow dataset  $\tilde{S}_i$  such that  $S_i \cap \tilde{S}_i = \tilde{I}_i$ , a membership collisions label for each data  $\mathbf{x}'_i$  will be  $y'_i = \mathbb{1}(\mathbf{x}'_i \in \tilde{I}_i)$ .

In Step 6, attack model  $f(\cdot)$  outputs the predicted probability about whether a synthetic data is colliding member. Adversaries then expose a data set  $R_A$  that with high prediction scores.

For attack model (2) (unlimited synthetic data) such that  $N_s > 1$ , the adversary repeat the Step1 to Step 4  $N_s$  times and gets  $N_s$  labeled shadow datasets  $\{\tilde{S}_1, \tilde{S}_2, \dots, \tilde{S}_{N_s}\}$  such that each of them with size  $N_s \times |D_t|$ . Then the adversary concat (" $\parallel$ ") all shadow datasets together to train the attack model.

Note that in the worst-case (to the adversary), where the intersection between the training set and the synthetic dataset could be empty, the adversary of *TableGAN-MCA* cannot recover anything

from the private training data. To avoid such a case, we would discretize the synthetic dataset to generalize the range of each feature such that there is a non-empty intersection. In this way, we could (at least) recover coarse-grained information regarding the members within the training data. We show the details of the discretization operation in Section 5.1.

## 5 EVALUATION

In this section, we first introduce the methods of tabular data synthesis, then introduce the evaluation metrics. Next we show the attack performance of *TableGAN-MCA* as well as the comparisons with recent works.

### 5.1 Dataset Synthesis

We perform experimental evaluations on three commonly used [3, 8, 35, 40, 46] real-world tables, Adult [39], Lawschool [38] and Compas [22].

**Adult:** The US Adult Census dataset is a repository of 48842 entries extracted from 1994 US Census dataset, where 45222 entries have complete information. After pre-processing, it remains 1 numerical feature, 12 categorical features and 1 binary label.

**Lawschool:** This dataset comes from the Law School Admission Council’s National Longitudinal Bar Passage Study. It contains application records for 25 different law schools with 86022 individuals. It has 2 numerical features, 5 categorical features and 1 binary label.

**Compas:** COMPAS recidivism risk score and criminal history data is collected by ProPublica in 2016. After pre-processing, it remains 5278 entries with 4 numerical features, 6 categorical features and 1 binary labels.

Note that unlike MIAs attacking classifiers that produce predicted labels with probability, generative models only output synthetic samples. The labels in generated datasets serve as an ordinary feature like other features when training attack models. Therefore, for simplicity, we use the three binary-labeled datasets in our experiments.

**Tabular data synthesis.** For training generative models, we apply Tabular Variational Autoencoder (TVAE) [46], CTGAN [46], WGAN-GP [15] and WGAN-WC [2] for their superior modeling quality in tabular synthesis. To facilitate data synthesis, we have the following additional data pre-processing. (1) We discretize imbalanced and sparse numerical values in given columns to categorical values. (2) We normalize numerical columns into (0, 1) or (−1, 1). (3) We one-hot encode all categorical features (4) We split the dataset into the training set ( $D_t$ , 70% records) and test set ( $D_s$ , 30% records) (see row 1 and row 2 in Table 3). The training set is used for dataset synthesis and the test set is used for examining the utility of the synthetic data.

**Discretization in pre-processing.** Features in tabular dataset are either categorical or numerical variables. Unlike pictures, some numerical columns are non-Gaussian distribution, that is, it either has long tails, sparse distribution or multiple modes. Generative models cannot model them well without appropriate pre-processing. To address this issue, we discretize the imbalanced and sparse numerical values to categorical values. In the experiments, such simple discretization in pre-processing exhibits decent performance in generating complex features while keeping original statistics. Note

**Table 3: Dataset Statistics for GAN synthesis.**  $\Pr[\#x = 1]$ : unique training data proportion;  $\Pr[\#x \leq 3]$ : Proportion of training data with frequency less than 3.

	Adult	Lawsch	Compas
# of Train $D_t$ (70%)	31655	60215	3694
# of Test $D_s$ (30%)	13567	25807	1584
$\Pr[\#x = 1]$	79.39%	71.28%	63.72%
$\Pr[\#x \leq 3]$	86.07%	81.85%	74.28%

that discretization definitely makes some records of the original dataset share the same values (similar to  $k$ -anonymity [45]). We show the uniqueness of the records after pre-processing in Table 3 (row three and four), where a large proportion of sensitive data points can still be uniquely identified before feeding into generative models.

### 5.2 Metrics

**5.2.1 Data Utility Metrics.** For data utility evaluation, we consider two measurements: machine learning efficacy (models trained on a synthetic dataset and the original dataset provide similar predictions) and distribution fitness (a synthetic dataset is statistically similar to its original dataset in all attributes).

For distribution fitness, we present 1-way marginals that are approximated by the Empirical Cumulative Distribution Function (ECDF) for each attribute. Having ECDFs of real and synthetic data, we compute attribute-wise Wasserstein distance, i.e.,  $l_1(x_i, x'_i) = \int_{-\infty}^{+\infty} |U_i - V_i|$ , where  $U_i$  and  $V_i$  are respective CDFs of real attribute  $x_i$  and synthetic attribute  $x'_i$  [37]. We compare the expected value of ECDFs by  $\mathbb{E}_i(l_1) = \frac{1}{n} \sum_{i=1}^n \{l_1(x_i, x'_i)\}$ .

**5.2.2 Attack Performance Metrics.** To evaluate the privacy of the released synthetic table, we consider membership collisions privacy, i.e., the *TableGAN-MCA* effect. We use precision and recall to evaluate the attack performance (following Shokri et al. [43]), since the synthetic dataset that is used to inference has a skewed label distribution. Specifically, precision measures the probability of an entry inferred as a member is indeed the member of the training dataset, denoted as  $\Pr(y = 1|\hat{y} = 1)$ . Intuitively, it implies the confidence of the attacker in guessing positive membership. Recall measures the probability of a member is correctly inferred as a member by the attacker, denoted as  $\Pr(\hat{y} = 1|y = 1)$ . It reflects the percentage of positives exposed in the attack. In evaluation, we report precision and recall by Precision-Recall (PR) curve since it is more informative than ROC-curve under the case of skewed label distribution [10]. A higher Area under the PR-curve (AUPRC) implies both higher precision and recall, and thus they are used to compare the attack efficacy.

In addition to the attack precision and recall, we also consider a recovery rate because it reflects what the proportion of training data  $D_t$  are being exposed to *TableGAN-MCA*. Let  $R_{\mathcal{A}}$  be recovered training data sets of the attack algorithm  $\mathcal{A}$ . The recovery rate  $\rho_{\mathcal{A}}$  of  $\mathcal{A}$  is defined as below:

$$\rho_{\mathcal{A}} = |R_{\mathcal{A}}|/|D_t|. \quad (5)$$

Note that the recovery rate shares the same numerator as the recall of the attack model  $f(\cdot)$  but the different denominator ( $|D_t|$  vs  $|I|$ ).



**Table 4: Model prediction accuracy (%) trained on real training  $D_t$  (“Base”) and GAN-synthesized datasets  $S$ .  $\mathbb{E}_i(l_1)$  denotes the average of all attribute-wise Wasserstein distance.**

	Methods	DT	MLPC	Ada	LR	$\mathbb{E}_i(l_1)$
Adult	Base	85.39	84.11	86.28	84.74	0
	TVAE	79.24	77.53	78.72	80.2	0.0207
	CTGAN	81.53	81.76	82.3	82.41	0.0266
	WGANWC	82.74	83.62	84.16	83.96	0.0075
	WGANGP	83.16	83.95	<b>84.24</b>	83.93	<b>0.0039</b>
Lawschool	Base	81.90	89.54	87.23	87.68	0
	TVAE	79.53	85.38	85.17	85.26	0.0120
	CTGAN	76.35	80.91	81.02	81.37	0.0283
	WGANWC	77.54	80.76	80.56	80.93	0.0073
	WGANGP	80.14	86.10	86.02	<b>86.79</b>	<b>0.0047</b>
Compas	Base	69.89	70.58	71.65	71.46	0
	TVAE	64.42	68.07	64.33	68.33	0.0159
	CTGAN	58.14	60.21	59.5	58.12	0.0373
	WGANWC	65.34	66.5	65.06	<b>68.46</b>	<b>0.0095</b>
	WGANGP	64.12	66.91	65.20	68.34	0.0179

### 5.3 Synthetic Data Utility

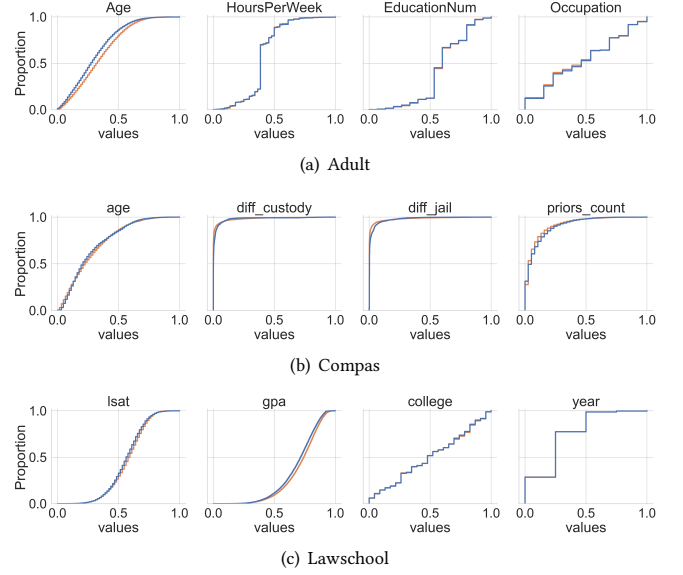
We evaluate machine learning efficacy of synthetic data generated by four generative models, CTGAN [46], TVAE [46], WGAN-GP and WGAN-WC vary four binary classifiers: DecisionTreeClassifier, MLPClassifier, AdaBoostClassifier, and LogisticRegression (Standard scikit-learn machine learning library, see middle columns in Table 4). We also compare ECDFs using the average of all attribute-wise Wasserstein distance  $\mathbb{E}_i(l_1)$  (see the last column in Table 4). All numerical features are min-max scaled to (0, 1) and categorical features are one-hot encoded before feeding into the classifier. For “base”, we trained on the sensitive dataset  $D_t$  that is used for data synthesis and test on the real test set  $D_s$  (see Table 3). For synthetic data, we trained on a synthetic dataset  $S$  of the same size as the sensitive dataset and test on the same real test set  $D_s$ . To implement CTGAN and TVAE, we directly feed our pre-processed data into the module CTGANSynthesizer and TVAESynthesizer of the SDGym [5] (published code for [46]).

According to Table 4, the synthetic dataset generated by WGAN-GP, WGAN-WC, CTGAN and TVAE can greatly restore the prediction ability of the model trained on original dataset. TVAE is least ideal than the others in the Adult dataset. CTGAN is least ideal than the others in the Compas dataset. We will use these learned generative models to perform *TableGAN-MCA* experiments later.

For marginal fitness, we depict an additional ECDF comparison between real and synthetic Adult, Lawschool and Compas datasets generated by WGAN-GP in Fig. 5. In our experiments, we depict ECDFs of continuous variables (i.e., age, isat) and more complex categorical variables (i.e., hours per week, priors count) since they are more difficult to fit. As can be seen in Fig. 5, the marginals of the synthetic dataset are almost indistinguishable from the original one, thus supporting any statistical queries.

### 5.4 Attack Performance

**5.4.1 Performance Evaluation on TableGAN-MCA.** In this section, we evaluate *TableGAN-MCA* of Alg. 1 on the Adult, Lawschool and Compas datasets. The training and inference data statistics of *TableGAN-MCA* are presented in Table 5, where positive percentage implies the membership collision proportion. Both target models and shadow models are WGAN-GP. The attack model is trained on the shadow dataset  $\tilde{S}$  and tested on the synthetic dataset  $S$ .



**Figure 5: The Empirical Cumulative Distribution of each attribute in the Adult, Compas and Lawschool datasets (Orange line for real and blue line for synthetic).**

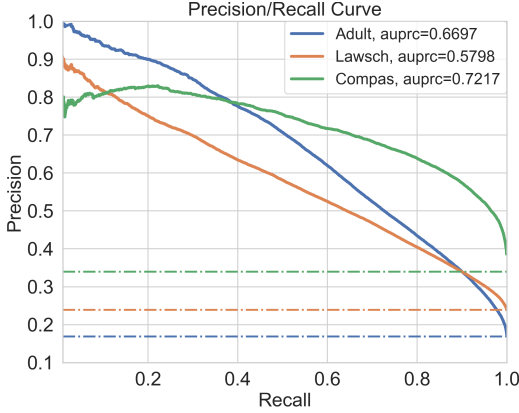
**Table 5: Training and inference statistics for the Adult, Compas and Lawschool datasets in *TableGAN-MCA*.**

	Adult	Lawsch	Compas
$ \tilde{S} $ (Train)	31655	43011	3694
$ S $ (Inference)	31655	43011	3694
$\Pr_{\tilde{S}}[y_i = 1]$	15.99%	22.68%	40.49%
$\Pr_S[y_i = 1]$	16.90%	23.89%	34.00%

*TableGAN-MCA* provides a promising attack against the GAN-synthesized tables. We report the PR-curve of the attack model in Fig. 6 when  $N_s = 1$ , i.e.,  $|S| = |D_t|$ . In Fig. 6, PR-curve reflects the trade-off between precision and recall for different probability thresholds  $T$ . Particularly, after providing the inference data (the released GAN-synthesized tables) to the *TableGAN-MCA* attack model, we receive a set of probabilities for each record of the test data that predicts whether a record is a member.

As illustrated in Fig. 6, we find that by setting a suitable threshold  $T$ , the adversary can expose approximate 30% colliding members with confidence over 83.91%, 69.40% and 81.24% for the Adult, Lawschool and Compas datasets, respectively. This means that the adversary significantly increases its ability to assert that these entities are members. Furthermore, when setting confidence to 80%, we have 36.1%, 12.7%, 36.5% positive percentages being exposed, which correspond to 1931, 1304 and 458 individual’s sensitive entries in the Adult, Lawschool and Compas datasets, respectively. According to Fig. 6, we list *TableGAN-MCA*’s recovery rates (Eq.(5)) with different precision configurations in Table 6.

**Adversary’s knowledge enhances the attack performance of *TableGAN-MCA*.** Fig. 7 reports the PR-curve and AUPRC of *TableGAN-MCA* when  $N_s = 10$ . That is, the adversary has multiple copies of the released synthetic data. In particular, when  $N_s = 10$ , the adversary trains 10 independent shadow GAN for each one of



**Figure 6: Attack effect of *TableGAN-MCA*. The dash-dot lines imply random guess baselines (0.1690, 0.2389, 0.3400 for Adult, Lawschool and Compas datasets, respectively.)**

**Table 6: *TableGAN-MCA*’s recovery rate  $\rho_{\mathcal{A}}$ . ( $|\mathcal{R}_{\mathcal{A}}|$ : # of recovered data points under attack algorithm  $\mathcal{A}$ )**

Datasets	$\rho_{\mathcal{A}}(\%)$	$ \mathcal{R}_{\mathcal{A}} $	$ D_t $	Precision	Recall
Adult	3.04	962	31655	0.9	0.16
	6.10	1931	31655	0.8	0.36
Lawsch	3.03	1305	43011	0.8	0.13
	4.66	2003	43011	0.75	0.18
Compas	12.41	458	3694	0.8	0.37
	17.17	634	3694	0.75	0.43

$S_i$  and finally obtains a shadow dataset  $N_s^2$  times the size of  $|D_t|$ . In Fig. 7, the performance of *TableGAN-MCA* is greatly improved by increasing the number of synthetic copies  $N_s$ . We also show the PR-curve comparison for the three datasets in Fig. 7. That is, given 30% recall, 10 copies boost the precision from 86.81% ( $N_s = 1$ ) to 90.70% ( $N_s = 10$ ). Given 90% precision, the recall is boosted from 22.25% ( $N_s = 1$ ) to 33.01% ( $N_s = 10$ ). We conclude that more copies of the synthetic data allow the adversary to make better approximation to generated distribution  $P_g$ , and thus generate more informative labeled shadow samples to train the attack model.

**TableGAN-MCA achieves commendable attack performance even with fewer synthetic queries.** When  $N_s = 0.25$  (an adversary queries the target Generator  $0.25 * |D_t|$  times), *TableGAN-MCA* achieves 0.6674 AUPRC ( $N_s = 1$  is 0.6697), and recovers 1,409 data points ( $N_s = 1$  is 1565) under 75% precision in the Adult dataset. More details are shown in Appendix B.

**The generation quality of the target/victim model positively impacts attack performance.** Fig. 8 depicts *TableGAN-MCA*’s performance on four different target/victim models: WGAN-GP, WGAN-WC, CTGAN and TVAE. The shadow model in use is exactly the same as target models. Combining the results of Fig. 8 and Table 4, we conclude that target/victim generators with high generation quality often attain high attack performance. For instance, TVAE with the lowest prediction accuracy score in prediction accuracy (see Table 4) also achieves unsatisfactory performance in *TableGAN-MCA* on the adult dataset. This echoes what CTGAN performs in the Compas dataset. Additionally, we observe that

attack performance of TVAE is more sensitive to its generation quality.

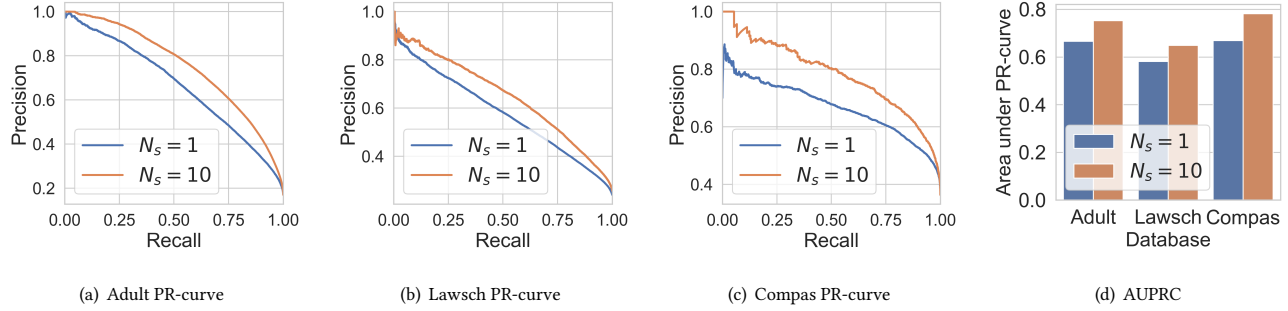
**The type of shadow models has limited impact on attack performance.** Note that the adversary may have no knowledge about the structures and parameters of target/victim generative models. Fig. 8(b) compares the attack performance by using four different shadow models (WGAN-GP, WGAN-WC, CTGAN and TVAE) to attack target WGAN-GP. As can be seen, various shadow model attacks (“wganwc”, “ctgan”, “tvae”) work as well as the identical shadow model attack (“wgan-gp”). TVAE shadow models perform worst, in large part due to its poor learning ability in the Adult dataset.

**The success of *TableGAN-MCA* is mainly due to the observed collision,** the membership collisions indicator and the shadow model in use. In particular, the collision between synthetic data and training set provides the opportunity for recovering training data. The membership collisions indicator, which captures the statistical patterns behind colliding members, guarantees more accurate and informative features for training the attack model. The shadow model in use provides enough labeled data to train the attack model so as to learn from the statistical patterns of the colliding members.

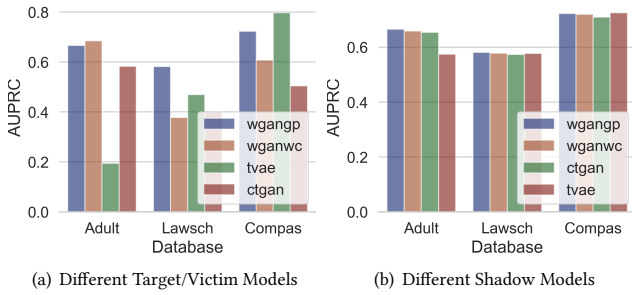
**Attack scalability.** The key to the success of the *TableGAN-MCA* is the possibility of collisions between raw training datasets and synthetic datasets. For real-world tabular data, attributes usually have a finite domain range. Hence, the dataset dimension indicates its overall domain range. Namely, low-dimensional tables are more likely to incur sample collisions when the generator creates those synthetic tables. Therefore, *TableGAN-MCA* discovers additional privacy risks – membership disclosure via collision attacks – for low-dimensional data. *TableGAN-MCA* potentially fits high-dimensional data if adversaries reduce the data granularity by generalizing attributes. The *TableGAN-MCA* works since the synthetic data would have a higher chance to collide with the training datasets. In our experiments, *TableGAN-MCA* achieves 0.871 AUPRC by bucketizing the “age” attribute in synthetic Adult datasets into 10 bins (no-bucketization baseline: 0.668).

**5.4.2 Comparisons between *TableGAN-MCA* and existing MIAs.** Firstly, ***TableGAN-MCA* recovers member data points from the GAN-synthesized tables previously assumed to be resilient to table-GAN [35].** We evaluate the performance of table-GAN against the same WGAN-GP that used in *TableGAN-MCA* evaluation. Notice that we test their MIAs directly on the target discriminator instead of the shadow discriminator due to the fact that if the target discriminator fails, the shadow model will perform even worse. We report the accuracy of membership prediction (member/non-member) of table-GAN, which are 50.17%, 50.80% and 50.67% for Adult, Lawschool and Compas datasets, respectively (50% is the baseline of random guess). Taken altogether the experiment results in Fig. 6, we conclude that GAN APIs with a black-box access assumed to be resilient to table-GAN (targeting on a discriminative model) [35] may still disclose partial sensitive training information under *TableGAN-MCA*.

Secondly, **the MIAs proposed in GAN-Leaks and LOGAN cannot disclose membership collisions.** Note that the existing MIAs against GANs may work in the MCA scenario. Thus, we perform additional experiments to infer membership collisions of each



**Figure 7: Attack performance comparisons between one synthetic copy ( $N_s = 1$ ) and ten synthetic copies ( $N_s = 10$ ) in *TableGAN-MCA*. (a), (b), (c): PR-curve comparison for three datasets; (d): AUPRC comparisons for three benchmarks.**



**Figure 8: Attack effect of *TableGAN-MCA* under different target/victim-shadow model settings. In (a), shadow models are the same as target/victim models.**

synthetic data point using their methods. In particular, we evaluate LOGAN (black-box attack with no auxiliary knowledge) and GAN-Leaks (full black-box generator attacks) under threat model (1) (given one copy synthetic data, see details in Section 3.2) and report the result in Table 7. MC and table-GAN are not included in this experiment. The reason is two-fold. First, the distance function of MC is not directly applicable to non-image datasets. Second, table-GAN requires predicted probability vectors of the target discriminator, which is not permitted in our threat model. Note that the synthetic dataset  $S$  has imbalanced membership collisions labels (Row 1 in Table 7) that are different from Shokri’s shadow model MIA [43] (random observation with 50% real members) since the number of colliding data points (members) is usually unequal to non-colliding ones (non-members).

We observed that the results in GAN-Leaks are close to the random guess baseline. This is due to the reconstruction loss  $L(x, x^*) = 0$  for all synthetic data regardless of membership collisions (the optimal reconstruction of a synthetic data  $x$  is itself). Furthermore, LOGAN did not show convincing inference results since it never learns the intersection between the synthetic data and the private training data. In comparison, *TableGAN-MCA* learns such an intersection (by which we recover partial training data) through the intersections of the published synthetic data (by mimicking the private training data) and shadow (synthetic) data (by mimicking the original synthetic data).

**Table 7: The attack AUPRC comparison (mean  $\pm$  SD). Base implies random guess baseline. We use WGAN-GP as target/victim as well as shadow models.**

	Adult	Lawschool	Compas
Base	0.1690 $\pm$ 0.0038	0.2389 $\pm$ 0.0067	0.3400 $\pm$ 0.0233
LOGAN	0.2237 $\pm$ 0.0194	0.2512 $\pm$ 0.0172	0.3154 $\pm$ 0.0343
GAN-Leaks	0.1667 $\pm$ 0.0063	0.2514 $\pm$ 0.0061	0.3256 $\pm$ 0.0301
Proposed	0.6681 $\pm$ 0.0348	0.5805 $\pm$ 0.0144	0.7228 $\pm$ 0.0556

In summary, the MIA classifiers that identify membership fail to identify those membership collisions since the decision boundaries of our attack classifier is different from those of MIAs against GANs.

## 6 TABLEGAN-MCA ANALYSIS

In this section, we discuss the factors that may impact the attack performance of *TableGAN-MCA* from the following aspects, such as GAN training set size, GAN training epochs and GAN training data frequencies. We choose WGAN-GP as targets as well as shadow model for its superior modeling quality and stability in *TableGAN-MCA* experiments.

### 6.1 GAN Training Set Size

**The size of the training dataset for a GAN model positively impacts the attack performance.** Fig. 9 depicts the positive impact of training dataset size on prediction accuracy and AUPRC of *TableGAN-MCA*, where 1.0 in x-axis indicates the full size of a given dataset,  $N_s = 1$ . Especially, when the size of the training dataset is less than 0.5 of the full dataset, increasing the size has a significant impact on the attack performance. The intuition behind the experimental results is two-fold. First, less training data decrease the number of colliding members (positives) in fixed amount of synthetic datasets thus decreases the attack effect. Second, GAN learns a less accurate data distribution if trained on a smaller dataset. Synthetic data generated by such a distribution contain less information than the original training data hence hard for the adversary to learn the statistical patterns of the members/non-members. Note that our results do not conflict with [7, 20, 27] since we use different measurements (PR space vs ROC space) that focus on different domains [10]. Additionally, our attack target (test data) is also different. We aim to recover the colliding member data from the released synthetic dataset whereas they aim to infer the membership of



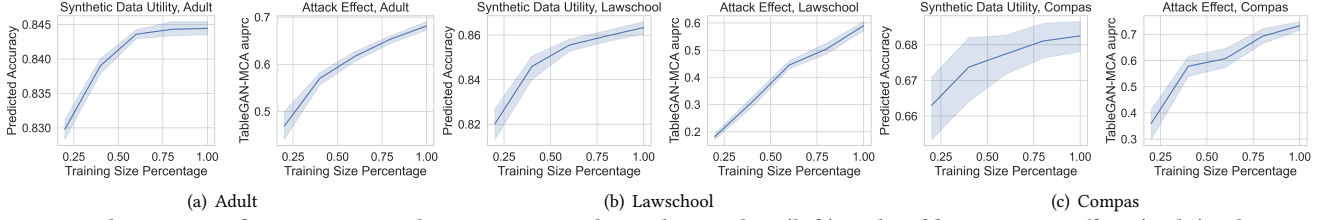


Figure 9: The impact of GAN training data size on synthetic data utility (left) and *TableGAN-MCA* effect (right). The x-axis indicates the amount of GAN training data

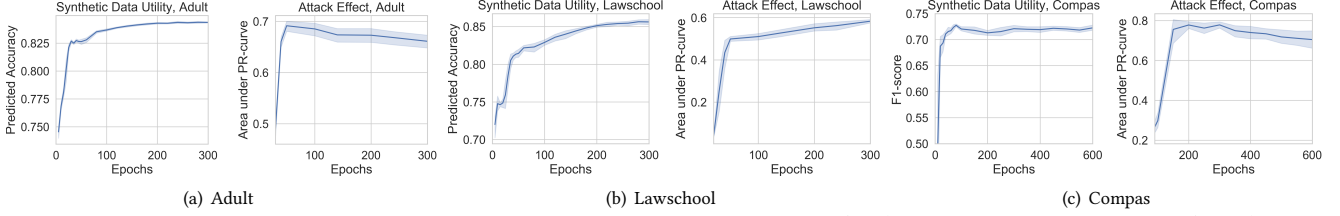


Figure 10: The impact of GAN training epochs on Synthetic data utility (left) and *TableGAN-MCA* effect (right).

a random target data point, and thus we learn different decision boundaries.

## 6.2 GAN Training Epochs

Epochs impact the attack performance of *TableGAN-MCA* by impacting the knowledge learned by GAN models. We study the attack performance on different training stages by setting different epochs in Fig. 10, where we report the attack prediction accuracy and attack AUPRC.

As seen from Fig. 10, we find that the membership leakage starts at the very beginning of the training epoch, even before the GAN reaches the Nash equilibrium. Interestingly, in Adult and Compas, the attack effect seems to slightly decrease when we set a larger epochs for training GAN models. Since *TableGAN-MCA* tends to recover the data with high appearance frequency (recall Fig. 3), we conclude that with increasing epochs, GAN models learn more about the training data distribution; hence, the released synthetic data contain more information, which enhances the attack performance. However, once the GAN models learn the details of the data distribution, such details about the distribution would dilute the frequency of those data supposed to have high frequency. The attack performance of *TableGAN-MCA* is then potentially dropped.

## 6.3 Training Data Frequencies

Training data frequencies are positively correlated with training data recovery probabilities by *TableGAN-MCA*. We first compute the recovery possibility and appearance frequency for each data point. We then plot the recovery possibility over the values of data points frequency in Fig. 11. For each dataset, we set two precision-scores of *TableGAN-MCA* and plot the training data frequency-recovery rate curves. Overall, highly frequent training data are more susceptible to *TableGAN-MCA*. For instance, when attacking Adult datasets with 80% precision, 41.5%(784/1892) training data with appearance more than three times are recovered by *TableGAN-MCA* whereas only 0.6%(510/25130) of unique training data ( $\#x = 1$ ) are recovered by *TableGAN-MCA*.

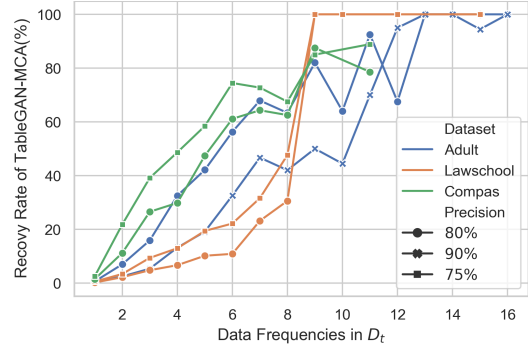


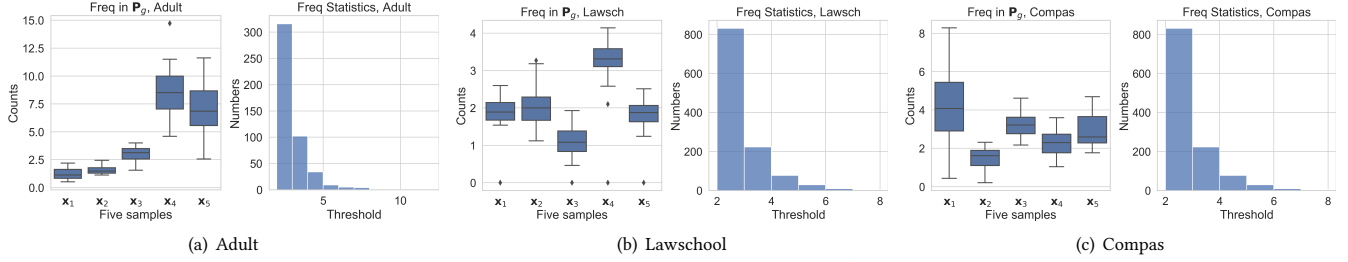
Figure 11: The impact of training data frequencies on *TableGAN-MCA* effectiveness. The attack precision is set to be one of  $\{75\%, 80\%, 90\%\}$ .

For highly frequent training data, GANs inevitably learn and output these common representations frequently; thus it is easy to recover such highly frequent data by *TableGAN-MCA*. The re-identification threats of these data caused by *TableGAN-MCA* are limited since each of them correspond to several individuals and lack of uniqueness.

Unique training data, on the other hand, have more risks for being linked to specific people once recovered by *TableGAN-MCA*. Therefore, it deserves further exploration for the reason of being exposed.

Generalization of GAN models may accidentally increase the appearance of some unique data points in the synthetic data, therefore increasing their probability to be recovered by *TableGAN-MCA*. Since the *TableGAN-MCA* is based on data density in modeled distribution  $P_g$ , for a recovered unique training data point  $x_i$ , we study how *TableGAN-MCA* is impacted by the difference between data density of  $x_i$  in the training distribution  $P_r$  and that of the modeled distribution  $P_g$ .

According to our experiments, we discover that some unique training data ( $\forall x \in P_r, \#x_i = 1$ ) have unexpected high exposure in modeled distribution  $P_g$ . For example, in Fig. 12, we illustrate



**Figure 12: Five synthetic samples' count estimation in generated distribution (left) and statistics of data points that increase the frequencies in  $P_g$  across three datasets (right).**

the average counts (from 100 synthetic datasets following modeled distribution  $P_g$ ) of five data points that appear in the training dataset only once. As we can see, these five data points have higher counts than what they have in the training dataset ( $= 1$ ). Such an observation indicates that the generator of GAN models unfairly increases the probability of exposure of some data points under *TableGAN-MCA*. We also find that such an observation is not rare. For instance, according to the statistics in Fig. 12 (Adult), roughly 470 (1.5% of the training dataset) unique entries at least double their exposure; roughly 150 (0.47% of the training dataset) unique entries at least triple their exposure.

Next, we explore the factors that potentially trigger our observations by a set of experiments inspired by unintended memorization [6]. Specifically, unintended memorization identifies the impact of the presence of one training input on the modeled distribution  $P_g$  learned by GAN. Note that this experiment resembles the definition of differential privacy (DP) [12]. DP is more generic and rigorous as it measures that the probability difference varies all possible functions and all data points, which is computationally infeasible in our measurements. In this case, we narrow down the design by observing the difference between a sample density in two generated distributions  $P_g$  trained on neighboring training sets.

Let  $D_t$  be the sensitive training set,  $x_i \in D_t$  be a target data point and  $D'_t = D_t \setminus x_i$  be the neighboring dataset such that the Hamming distance  $d_H(D_t, D'_t) = 1$ . Let  $G$  be a learned generator trained on  $D_t$  and  $G'$  be a generator trained on  $D'_t$ . We measure the difference between the probability of producing a synthetic data point  $x_i$  with (prior) and without (posterior) the input data point  $x_i$ .

$$\frac{\Pr(G(z) = x_i | D_t)}{\Pr(G'(z) = x_i | D'_t)} \quad (6)$$

Following a recent work [6], GAN models do not memorize a data point if it does not exist in the training dataset  $D_t$ . Thus, if Eq. (6) approaches 1, we say that the target data  $x_i$  is unlikely to be memorized by the GAN. The pseudo-code of the experiment is presented in Alg. 2. In the experiment, we use 20 different GANs ( $N_k = 20$ ) and some of target data to estimate Eq. (6). We report the experimental results of five target data ( $N_c = 5$ ) in Fig. 13.

From Fig. 13, we choose the same samples (data points) as in Fig. 12 to compare how prior (with a target  $x_i$ ) and posterior densities (without target  $x_i$ ) differ in modeled distribution. We find that the presence of the target entry  $x_i$  has limited influence on its frequency in modeled distribution  $P_g$ . Even if some data point  $x_i$  is absent in the training set, its probability density in synthetic distribution  $P_g$  is still high, e.g.,  $x_4, x_5$  in the Adult dataset. This is perhaps

---

**ALGORITHM 2:** Memorization Experiment.

---

**Input:**  $\{x_1, \dots, x_{N_c}\}$ : sample data points;  $D_t$ : private training dataset.

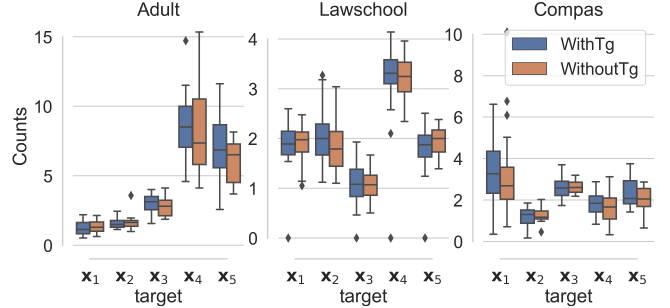
**Output:**  $\{\Pr[x_k | G_i(z)]\}$ : prior frequency;  $\{\Pr[x_k | G'_i(z)]\}$ : posterior frequency.

```

while  $k : 1 \rightarrow N_c$  do
  while  $i : 1 \rightarrow N_k$  do
    Generative model  $G_i \leftarrow$  Train on  $D_t$ ;
     $\Pr[x_k | G_i(z)] \leftarrow$  Estimate by Eq. (4);
     $D'_t \leftarrow D_t \setminus \{x_k\}$ ;
    Generative model  $G'_i \leftarrow$  Train on  $D'_t$ ;
     $\Pr[x_k | G'_i(z)] \leftarrow$  Estimate by Eq. (4);
  end
end
return  $\{\Pr[x_k | G_i(z)]\}, \{\Pr[x_k | G'_i(z)]\}$ 

```

---



**Figure 13: Memorization experiments on three datasets. The blue-color boxplot depicts the frequencies when the target entry is in the training set while the orange one depicts the frequencies when the target entry is deleted from the training set.**

because GAN's generalization smooths the sudden change that happened in the probability space of the training set. For instance, the density of the a target point  $x_i$  in  $P_r$  may be lower than the surrounding points, whereas the GAN smooths such sudden changes in the probability space, and thus it is unintended to increase its probability of exposure. In another aspect, such a rough probability space in real distribution may be attributed to insufficient sampling or unbalanced sampling. As such, cautious data collection may have positive impact in mitigating such influence. Understanding this complicated phenomenon with more explicit proof is our future work. Currently, we summarize that the unique training data recovered by *TableGAN-MCA* is mainly due to the GAN's generalization

rather than the unintended memorization. This result implies that mitigating the attack effect of *TableGAN-MCA* may inevitably compromise the availability of released synthetic datasets, since GAN generalization is closely related to its generation ability, which potentially impacts the quality of generated data.

## 7 MITIGATION

In this section, we evaluate the mitigation effects of differential privacy and two customized defense methods against *TableGAN-MCA*.

### 7.1 Differentially Private WGAN-WC

**Differentially Private WGAN (DP-GAN) only has acceptable trade-offs for larger privacy budgets, and may hardly eliminates *TableGAN-MCA* without compromise synthetic data utility.** Differential privacy [12] provides a quantified solution to output randomized answers. In this work, we apply a standard approach of differentially private iterative training procedure (DP-SGD, short for DP stochastic gradient descent) [1, 30] to the GAN to train a  $(\epsilon, \delta)$ -differentially private generator oracle. Otherwise, since DP-SGD perturbs the training process of discriminative models, such mitigation may achieve sub-optimal trade-offs between membership collision privacy and synthetic data utility. In the experiments, we implement the DP framework according to [30] and account the privacy budget  $(\epsilon, \delta)$  using RDP accountant released in Tensorflow/Privacy project. Note that WGAN-GP, TVAE and CTGAN do not have DP versions, and thus we study the DP version of WGAN-WC. The generation quality and *TableGAN-MCA* effect of non-private baseline are shown in Fig. 14 followed by Table 4 and Fig. 8.

To implement DP-WGAN, we train a differentially private discriminator. The generator is differentially private because of the post-processing [13]. We add calibrated noise into each gradient of the discriminator during training. The accumulation of multiple Gaussian noise addition [11] relies on privacy accountant techniques [1] and Rényi differential privacy [31]. We provide DP-related hyper-parameters in Table 8, Appendix C.1.

We provide the experimental results of the machine learning utility and *TableGAN-MCA* effect when sharing differentially private synthetic data in Fig. 14. The shadow GANs in use are non private WGAN-WC. The privacy budget  $\epsilon$  measures the amount of privacy leakage and a smaller value means more privacy-preserved.  $\delta$  denotes the probability of violating  $\epsilon$ -DP, which is set to  $\frac{1}{O(|D_t|)}$ . As can be seen from Fig. 14, the DP method has some positive effect in defending against the *TableGAN-MCA*. For Adult datasets, when privacy budget  $\epsilon \approx 2.0$ , the attack AUPRC decreases by 16.01% and model’s predicted accuracy decreases by 1.18% in comparison to the no-DP baseline (see dash dots in Fig. 14). For the Compas dataset, when privacy budget  $\epsilon \approx 8.0$ , the attack AUPRC decreases by 48.33% and model’s predicted accuracy decreases by 5.13% in comparison to the no-DP baseline. We also depict the ECDF comparison between the original training data and differentially private synthetic data for each marginal to show marginal fitness compromise in Fig. 18 (Appendix C.1). It is not surprising that DP-WGAN achieves sub-optimal trade-offs when protecting against *TableGAN-MCA*, since the memorization experiment shows that the presence

---

### ALGORITHM 3: GAN-constrained Training (Improved defense)

---

**Input:**  $D_t$ : private training data;  $N_g$ : number of discriminator iterations per generator iteration;  $m$ : batch size

**Output:** A Synthetic dataset  $S$

```

for each iteration do
  while  $i : 1 \rightarrow N_g$  do
    Sample  $\{x^{(i)}\}_{i=1}^m \sim P_r$ ;
    Sample  $\{z^{(i)}\}_{i=1}^n \sim P_z, n > m$ ; Choose  $m$  of  $n$  priors
     $\{z^{(i)}\}_{i=1}^m$  s.t.,  $G(z) \notin D_t$ 
    Compute loss, backward, update gradients;
  end
  Sample  $\{z^{(i)}\}_{i=1}^n \sim P_z$ ; Choose  $m$  of  $n$  priors  $\{z^{(i)}\}_{i=1}^m$  s.t.,
   $G(z) \notin D_t$ ;
  Compute loss, backward, update gradients;
end
 $S \leftarrow G(z), \text{ s.t., } G(z) \notin D_t$ ; ▷ Naive defense
return  $S$ 

```

---

of individuals does not significantly affect the generated distribution. The membership collisions information that we intend to infer is perhaps highly correlated to population statistics (attributes correlation), which will be preserved even under DP training.

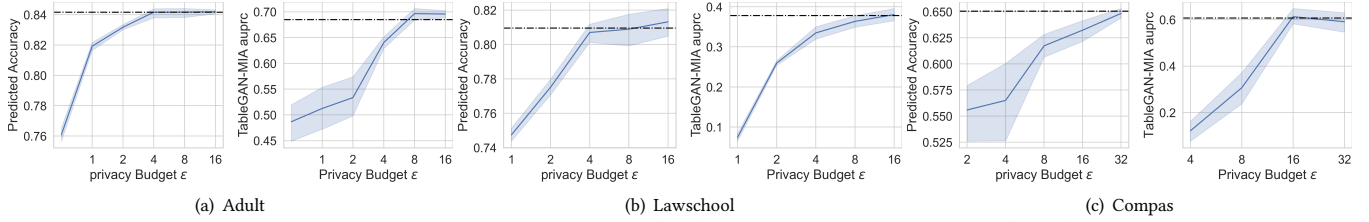
### 7.2 Customized Defense

**7.2.1 Remove Colliding Members. Removing colliding members protects against *TableGAN-MCA* but it reduces the distribution fitness.** The straightforward solution against *TableGAN-MCA* is to manually remove colliding members from the sampled synthetic dataset and share a cleaned version to the analysts (customers). The whole process is denoted as the “naive defense” (last steps in Alg. 3). We acknowledge the cleaned version can decrease the utility of original synthetic data, especially for distribution fitness. For example, we present the ECDF comparison of synthetic datasets generated by the naive defense (Fig. 15(b)) and no-defense (Fig. 15(a)). We show that the naive defense exhibits decreased marginal fitness compared with no-defense baseline. More ECDFs can be found in Figs. 19(a), 20(a), and 21(a) (Appendix C.2).

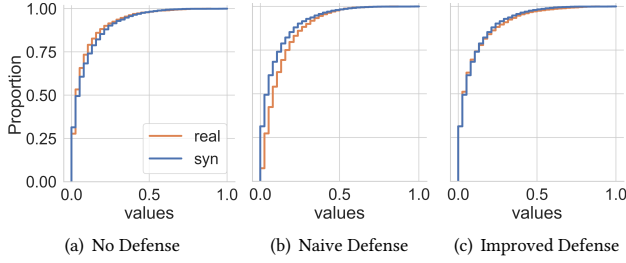
**7.2.2 GAN-constrained Training.** We propose a **GAN-constrained training technique**, to further improve synthetic data utility while protecting against *TableGAN-MCA*. This strategy is denoted as an “improved defense”. Simply put, we motivate GANs to generate a synthetic dataset  $S \sim P_g$  that is disjoint with the training set  $D_t$  while minimizing the distance between training data and generated data  $\mathcal{L}(D_t, S)$ , which is

$$S = \arg \min_{S_i} \mathcal{L}(S_i, D_t) |_{S_i \cap D_t = \emptyset}, \quad (7)$$

where  $\mathcal{L}$  denotes a distance metric. Since the discriminator of the WGAN minimizes the Wasserstein distance, we additionally add a constraint during training to force each sampled batch of the generator to be disjoint with  $D_t$ . To do so, we remove the intersection between the sampled batch and the training set every iteration before computing the loss function (see Alg. 3). Thus, WGAN automatically searches for the best substitution for such colliding samples at training.



**Figure 14: Differential private GAN-synthesized data utility (left) and TableGAN-MCA effect (right) for Adult, Lawschool and Compas benchmarks. Dash dot line denotes non-private WGAN with weight clipping baseline.**



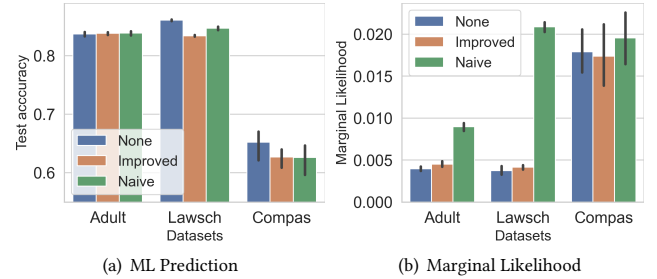
**Figure 15: ECDF comparisons for synthetic datasets generated by three methods. We choose “priors count” attribute in the Compas dataset.**

**7.2.3 Naive and Improved Defenses Evaluation. The improved defense in large part achieves superior trade-offs than the naive defense, and is almost comparable to the no-defense baseline.** We evaluate synthetic data utility of the naive defense, the improved defense and the no-defense (baseline) on WGAN-GP. Note that the baseline is vulnerable to TableGAN-MCA while naive and improved defenses protect against it. We evaluate machine learning efficacy in Fig. 16(a) and marginal fitness in Fig. 16(b).

In Fig. 16(a), we train machine learning models (Logistic Regression Classifier) on synthetic data sampled from the naive defense, the improved defense and the no-defense generator and predict on the real test data. Fig. 16(a) shows that synthetic data generated by naive and improved defenses achieve satisfying prediction accuracy on the Adult and Lawschool datasets. In the Compas dataset, mitigation methods decrease the prediction accuracy compared to the no-defense baseline.

In Fig. 16(b), we compare ECDFs using  $\mathbb{E}_i(l_1)$  (recall Section 5.2.1). The lower score implies better marginal fitness. The experimental result shows that the improved defense outperforms the naive defense, and is on par with the no-defense baseline. The improved defense succeeds in compensating the statistical deviation caused by the naive defense (see Fig. 15(c)). More ECDFs of the naive defense and the improved defense are shown in Figs. 19, 20, and 21.

In summary, both naive and improved defenses protect against TableGAN-MCA and in part preserve learning ability of released synthetic data. Moreover, the improved defense achieves better marginal fitness than the naive defense. Despite the potentially effective mitigation, TableGAN-MCA still remains a threat since the proposed defenses achieve sub-optimal privacy-utility trade-offs, eg, reduced synthetic data diversity, under-performance for tiny-domain datasets (see Compas datasets for details).



**Figure 16: Defenses comparisons on three datasets. Marginal Likelihood: compute  $\mathbb{E}_i(l_1)$ .**

## 8 RELATED WORK

Membership privacy is the existence of individuals [25, 36]. Existing studies show membership disclosure on discriminative machine learning models, e.g., classifiers [28, 33, 41–43, 47] and generative machine learning models, e.g., Generative Adversarial Networks [7, 20, 21, 35]. In the discriminative settings, an adversary infers whether a specific data point is used to train a target model by querying classifier APIs and using predicted probability vectors, labels, logits, etc., to train attack models. For instance, Shokri’s shadow model [43] infers membership against overfitted multi-class classifiers by training an attack model with labeled synthetic data, which mimic the private training data. Subsequent works further relax the adversary’s background knowledge [42] by extending attacks to the white-box [33] and the label-only settings [9, 26].

In the track of inferring membership against the generative models, there are several successful approaches, such as, tableGAN [35], LOGAN [20], MC [21] and GAN-leaks [7]. Note that some of these approaches is originally proposed against image data; however, they are possibly extendable to attack tabular data. That is, they are all related to this study. Hence, we briefly summarize these methods in this section. The conceptual comparisons are shown in Table 1. LOGAN [20] and table-GAN [35] leverage the output of the overfitting discriminator to train an attack model, which is a variant of Shokri et al. [43] in the context of GAN synthesis. However, their attacks require the predicted probability vector of the target discriminator at the inference phase (see column 3 in Table 1). In our experiment, we have already shown that a GAN resilient to their attacks may still expose training data to TableGAN-MCA. MC [21] and GAN-leaks [7] extract a customized membership indicator of an overfitting generator to train an attack model. We share a similar theoretical bases with theirs, that is, the modeled distribution of the generator behaves differently on training input versus the non-training one. However, our attack further recovers partial training



data by inferring membership of published synthetic data, which is out of their scope (see Columns 4 and 5 in Table 7). In this work, we empirically show that the membership inference classifier cannot be directly used to identify membership collisions in our attack model (see Table 7). Compared to those works, we propose a novel attack model, *TableGAN-MCA*, that exposes partial training data by exploiting the weakness of tabular data synthesis. Even though we share similar ideas with MIAs in generative setting, the attack model of *TableGAN-MCA* learns different decision boundaries. According to the experimental results, the success of the proposed attack relies more on population knowledge than individual presence, which is different from MIAs.

## 9 CONCLUSION

GAN-synthesized table releasing provides unprecedented opportunities for private data sharing that aims to study the regular pattern of population. In this work, we propose a novel membership collision attack, *TableGAN-MCA*, against the GAN-synthesized table. Our comprehensive experiments over the real-world datasets conclude some important findings. *TableGAN-MCA* achieves high recovering rate against the private training data from the published GAN-synthesized tables. Our in-depth studies suggest that the target model, training data size, training epochs and training data frequencies impact the attack performance of *TableGAN-MCA*. We further conclude that the training data leakage is mainly related to the published population statistics (attributes correlations), rather than the model memorization. To mitigate the effect of *TableGAN-MCA*, we find that differential privacy (applying DP-WGAN) does not show a satisfying result mainly due to the correlations between training data features. Based on our understanding on *TableGAN-MCA*, we propose two mitigation approaches, which substitute the published colliding members with similar non-private data entries. We hope that the concept of membership collisions defined and the attack methodology developed in this paper could inform the privacy community of such new potential leakage of data synthesis.

## ACKNOWLEDGMENTS

The authors, affiliated with Southeast University, were partially supported by Jiangsu Provincial Key Laboratory of Network and Information Security (No. BM2003201). Minhui Xue was, in part, supported by the Australian Research Council (ARC) Discovery Project (DP210102670). Aiqun Hu and Minhui Xue are the corresponding authors of this paper.

## REFERENCES

- [1] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep Learning with Differential Privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, Vienna Austria, 308–318.
- [2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein Generative Adversarial Networks. In *Proceedings of Machine Learning Research*, Vol. 70. PMLR, Sydney, Australia, 214–223.
- [3] Matias Barenstein. 2019. ProPublica’s COMPAS Data Revisited. *arXiv:1906.04711 [cs, econ, q-fin, stat]* (July 2019). <http://arxiv.org/abs/1906.04711>
- [4] Avrim Blum, Katrina Ligett, and Aaron Roth. 2013. A learning theory approach to noninteractive database privacy. *Journal of the ACM (JACM)* 60, 2 (2013), 1–25.
- [5] Sala Carles. 2019. SDGym Project. <https://github.com/sdv-dev/SDGym>
- [6] Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. 2019. The Secret Sharer: Evaluating and Testing Unintended Memorization in Neural Networks. In *28th USENIX Security Symposium (USENIX Security 19)*. USENIX Association, Santa Clara, CA, 267–284.
- [7] Dingfan Chen, Ning Yu, Yang Zhang, and Mario Fritz. 2020. GAN-Leaks: A Taxonomy of Membership Inference Attacks against Generative Models. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security (CCS 2020)*. ACM, New York, NY, USA.
- [8] Qingrong Chen, Chong Xiang, Minhui Xue, Bo Li, Nikita Borisov, Dali Kaarfar, and Haojin Zhu. 2018. Differentially private data generative models. *arXiv preprint arXiv:1812.02274* (2018).
- [9] Christopher A. Choquette Choo, Florian Tramer, Nicholas Carlini, and Nicolas Papernot. 2020. Label-Only Membership Inference Attacks. *arXiv:2007.14321 [cs.CR]*
- [10] Jesse Davis and Mark Goadrich. 2006. The relationship between Precision-Recall and ROC curves. In *Proceedings of the 23rd international conference on Machine learning - ICML '06*. ACM Press, Pittsburgh, Pennsylvania, 233–240.
- [11] Cynthia Dwork, Krishnamurthy Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. 2006. Our Data, Ourselves: Privacy Via Distributed Noise Generation. In *Advances in Cryptology - EUROCRYPT 2006*, Vol. 4004. Springer Berlin Heidelberg, Berlin, Heidelberg, 486–503.
- [12] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*. Springer, 265–284.
- [13] Cynthia Dwork, Aaron Roth, et al. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science* 9, 3–4 (2014), 211–407.
- [14] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*. 2672–2680.
- [15] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. Improved Training of Wasserstein GANs. In *Advances in Neural Information Processing Systems (NDSS)*. 5767–5777.
- [16] Anupam Gupta, Moritz Hardt, Aaron Roth, and Jonathan Ullman. 2013. Privately releasing conjunctions and the statistical query barrier. *SIAM J. Comput.* 42, 4 (2013), 1494–1520.
- [17] Anupam Gupta, Aaron Roth, and Jonathan Ullman. 2012. Iterative Constructions and Private Data Release. In *Theory of Cryptography*, Vol. 7194. Springer Berlin Heidelberg, Berlin, Heidelberg, 339–356.
- [18] Moritz Hardt, Katrina Ligett, and Frank Mcsherry. 2012. A Simple and Practical Algorithm for Differentially Private Data Release. In *Advances in Neural Information Processing Systems* 25. 2339–2347.
- [19] Moritz Hardt and Guy N. Rothblum. 2010. A Multiplicative Weights Mechanism for Privacy-Preserving Data Analysis. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science (Las Vegas, NV, USA)*. 61–70.
- [20] Jamie Hayes, Luca Melis, George Danezis, and Emiliano De Cristofaro. 2020. LOGAN: Membership Inference Attacks Against Generative Models. In *Proceedings on Privacy Enhancing Technologies*, Vol. 2019. 133–152.
- [21] Benjamin Hilprecht, Martin Härterich, and Daniel Bernau. 2019. Monte Carlo and Reconstruction Membership Inference Attacks against Generative Models. In *Proceedings on Privacy Enhancing Technologies*, Vol. 2019. 232–249.
- [22] Larson Jeff, Roswell Marjorie, and Atildakis Vaggelis. 2017. Compas Dataset. <https://github.com/propublica/compas-analysis>
- [23] Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).
- [24] N. Li, T. Li, and S. Venkatasubramanian. 2007. t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. In *2007 IEEE 23rd International Conference on Data Engineering*. 106–115.
- [25] Ninghui Li, Wahbeh Qardaji, Dong Su, Yi Wu, and Weining Yang. 2013. Membership privacy: a unifying framework for privacy definitions. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security - CCS '13*. Berlin, Germany, 889–900.
- [26] Zheng Li and Yang Zhang. 2020. Label-Leaks: Membership Inference Attack with Label. *arXiv:2007.15528 [cs, stat]* (July 2020).
- [27] Zinan Lin, Vyas Sekar, and Giulia Fanti. 2021. On the Privacy Properties of GAN-generated Samples. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 130)*, Arindam Banerjee and Kenji Fukumizu (Eds.). PMLR, 1522–1530. <http://proceedings.mlr.press/v130/lin21b.html>
- [28] Yunhui Long, Vincent Bindschaedler, Lei Wang, Diye Bu, Xiaofeng Wang, Haixu Tang, Carl A. Gunter, and Kai Chen. 2018. Understanding Membership Inferences on Well-Generalized Learning Models. *arXiv:1802.04889 [cs.CR]*
- [29] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkatasubramanian. 2007. L-diversity: Privacy beyond k-anonymity. *ACM Trans. Knowl. Discov. Data* 1, 1 (March 2007), 3–es.
- [30] H Brendan McMahan, Galen Andrew, Úlfar Erlingsson, Steve Chien, Ilya Mironov, Nicolas Papernot, and Peter Kairouz. 2018. A general approach to adding differential privacy to iterative training procedures. *arXiv preprint arXiv:1812.06210* (2018).

- [31] Ilya Mironov. 2017. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*. IEEE, 263–275.
- [32] Arvind Narayanan and Vitaly Shmatikov. 2008. Robust De-Anonymization of Large Sparse Datasets. In *Proceedings of the 2008 IEEE Symposium on Security and Privacy (SP '08)*. IEEE Computer Society, USA, 111–125.
- [33] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2019. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE, 739–753.
- [34] The Guardian Online. 2017. The Guardian view on Google's NHS grab: legally inappropriate. <https://www.theguardian.com/commentisfree/2017/may/17/the-guardian-view-on-googles-nhs-grab-legally-inappropriate>
- [35] Noseong Park, Mahmoud Mohammadi, Kshitij Gorde, Sushil Jajodia, Hongkyu Park, and Youngmin Kim. 2018. Data synthesis based on generative adversarial networks. *Proceedings of the VLDB Endowment* 11, 10 (2018), 1071–1083.
- [36] Atiqur Rahman, Tanzila Rahman, Robert Laganieri, Noman Mohammed, and Yang Wang. [n.d.]. Membership Inference Attack against Differentially Private Deep Learning Model. ([n.d.]), 19.
- [37] Aaditya Ramdas, Nicolas Garcia, and Marco Cuturi. [n.d.]. On Wasserstein Two Sample Testing and Related Families of Nonparametric Tests. ([n.d.]). <http://arxiv.org/abs/1509.02237>
- [38] Sander Richard, Knaplund Kris, and Winter Kit. [n.d.]. Law School Admissions. [http://www.seaphe.org/databases/FOIA/lawschs1\\_1.dta](http://www.seaphe.org/databases/FOIA/lawschs1_1.dta)
- [39] Kohavi Ronny and Becker Barry. 1996. UCI Machine Learning Repository. <https://archive.ics.uci.edu/ml/datasets/Adult>
- [40] Anian Ruoss, Mislav Balunović, Marc Fischer, and Martin Vechev. 2020. Learning Certified Individually Fair Representations. *arXiv:2002.10312 [cs, stat]* (Feb. 2020). <http://arxiv.org/abs/2002.10312> arXiv: 2002.10312.
- [41] Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, Yann Ollivier, and Hervé Jegou. 2019. White-box vs Black-box: Bayes Optimal Strategies for Membership Inference. In *Proceedings of Machine Learning Research (PMLR)*, Vol. 97. 5558–5567.
- [42] Ahmed Salem, Yang Zhang, Mathias Humbert, Pascal Berrang, Mario Fritz, and Michael Backes. 2019. ML-Leaks: Model and Data Independent Membership Inference Attacks and Defenses on Machine Learning Models. In *Proceedings of the 2019 Network and Distributed Systems Security Symposium*.
- [43] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, 3–18.
- [44] Theresa Stadler, Bristena Oprisanu, and Carmela Troncoso. [n.d.]. Synthetic Data – A Privacy Mirage. ([n.d.]).
- [45] Latanya Sweeney. 2002. k-ANONYMITY: A MODEL FOR PROTECTING PRIVACY. *Int. J. Unc. Fuzz. Knowl. Based Syst.* 10, 05 (Oct. 2002), 557–570.
- [46] Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. 2019. Modeling tabular data using conditional gan. In *Advances in Neural Information Processing Systems (NIPS)*. 7333–7343.
- [47] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. 2018. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*. IEEE, 268–282.

## APPENDIX

### A NETWORK STRUCTURE AND PARAMETERS

WGAN-GP, WGAN-WC shares the same network architecture. We set the Generator as Recurrent Neural Networks (RNNs). According to our experiments, the RNN has a positive effect on stabilizing the generator's outputs. Eq. (8) represents the Generator networks and Eq. (9) represents the Discriminator networks.

$$\begin{cases} h_1 = \text{ReLU}(\text{BN}(\text{FC}_{|z| \rightarrow 256}(z))) \\ h_2 = \text{ReLU}(\text{BN}(\text{FC}_{|z|+256 \rightarrow 256}(z \oplus h_1))) \\ G(\cdot)_{\text{con}} = \text{gumbel}_{0.2}(\text{FC}_{|z|+512 \rightarrow |r|}(h_2)) \\ G(\cdot)_{\text{cat}} = \tanh(\text{FC}_{|z|+512 \rightarrow 1}(h_2)) \end{cases} \quad (8)$$

$$\begin{cases} h_1 = \text{dropout}_{0.5}(\text{leakyReLU}_{0.2}(\text{FC}_{|r| \rightarrow 256}(r))) \\ h_2 = \text{dropout}_{0.5}(\text{leakyReLU}_{0.2}(\text{FC}_{256 \rightarrow 256}(h_1))) \\ D(\cdot) = \text{FC}_{256 \rightarrow 1}(h_2) \end{cases} \quad (9)$$

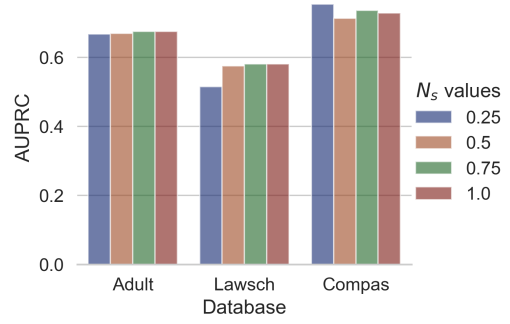


Figure 17: TableGAN-MCA performance when  $N_s \leq 1$ .

For TVAE and CTGAN, we apply the module CTGANSynthesizer and TVAESynthesizer of the SDGym [5]. Thus, the structures and hyper-parameters are exactly same as the originals' [46].

**Hyper-parameters.** For Adult and Lawschool datasets, we train 300 epochs and set batch size to 500. For Compas dataset, we train 600 epochs and set batch size to 100. Since the Compas dataset is much smaller than others, we find that less iterations could incur under-fitting. Additionally, balancing the number of D and G training sessions also helps to converge faster.

## B NUMBER OF SYNTHETIC QUERIES

We thoroughly discuss how the number of synthetic queries influences the attack performance and corresponding attack tricks.

### B.1 Limited Synthetic Queries

Many target model prediction APIs (MLaaS) implement a pay-per-query business model. Hence, reducing the number of synthetic queries saves the cost of performing TableGAN-MCA. However, a smaller synthetic dataset, having less membership collisions with the training dataset, decays the attack performance. To tackle this problem, we propose an approach that uses shadow data to fill up the synthetic data to match the size of the training set. That is, the adversaries obtain a synthetic dataset  $S$  of size  $0.25 * N$  by querying the target Generator. The adversaries then generate the shadow dataset of size  $|\bar{S}| = 0.75 * N$ . After that, the TableGAN-MCA adversaries attack  $S || \bar{S}$  instead of the original  $S$ .

We show the impact of a small  $N_s$  on TableGAN-MCA in Fig. 17. We find that few synthetic queries also yield decent attack performance. This resonates with the memorization experiment that the success of TableGAN-MCA is contingent more on basic data patterns.

### B.2 Unlimited Synthetic Queries

The TableGAN-MCA adversary continues to expose more training data when increasing the number of synthetic queries. In Fig. 7, we evaluate the TableGAN-MCA up to  $N_s = 10$ , which is not the ceiling of TableGAN-MCA capabilities. Due to computational constraints, we are limited to performing the attack up to  $N_s = 20$  and observe that the number of exposed training data of TableGAN-MCA is still increasing. This leaves open an interesting problem of whether the adversary could reconstruct the whole training dataset with unlimited queries.

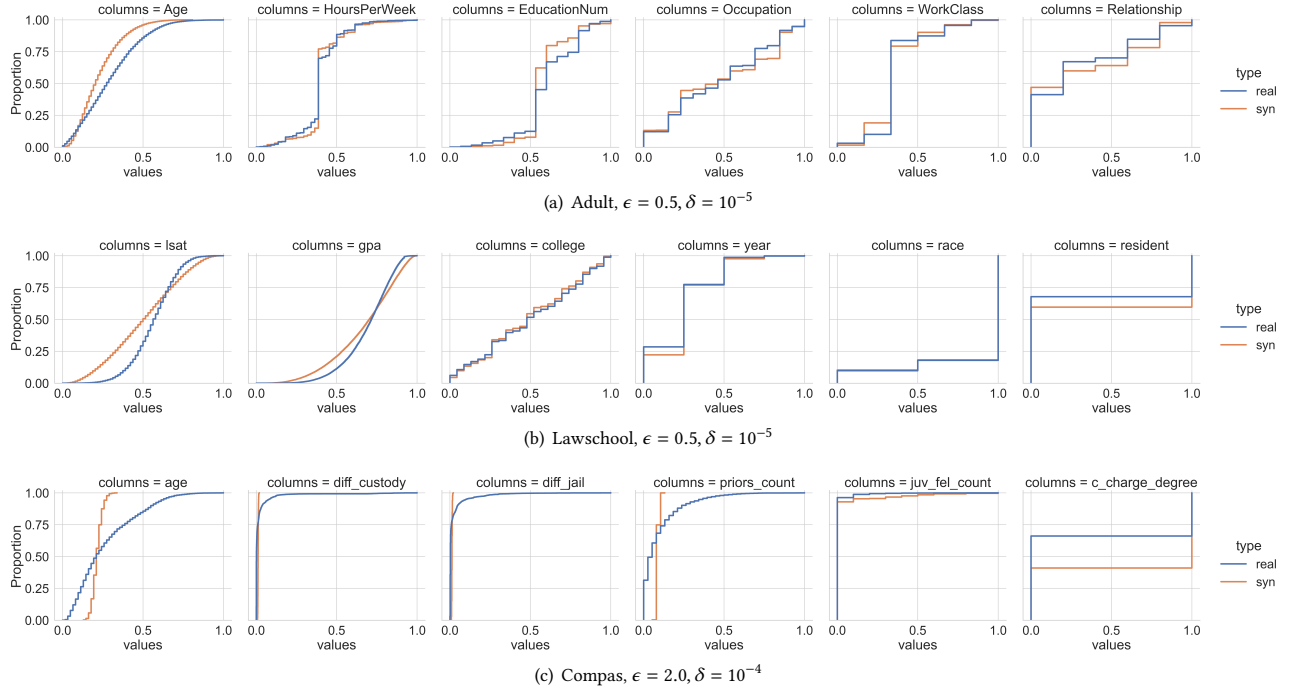


Figure 18: ECDF comparison between training data and differentially private GAN-synthesized data.

Table 8: Hyper-parameters in DP-WGAN.  $(\epsilon, \delta)$ : privacy budget;  $S$ : clip threshold;  $\sigma$ : standard deviation of the noise added in each step.

Datasets	$(\epsilon, \delta)$	$(S, \sigma)$	Sampling rate
Adult	$(0.5, 10^{-5})$	$(0.1, 0.5)$	500/31655
	$(1.0, 10^{-5})$	$(0.1, 0.45)$	500/31655
	$(2.0, 10^{-5})$	$(0.1, 0.4)$	500/31655
	$(4.0, 10^{-5})$	$(0.1, 0.3)$	500/31655
	$(8.0, 10^{-5})$	$(0.1, 0.17)$	500/31655
	$(16.0, 10^{-5})$	$(0.1, 0.11)$	500/31655
Lawschool	$(0.5, 10^{-5})$	$(0.1, 0.4)$	500/43011
	$(1.0, 10^{-5})$	$(0.1, 0.45)$	500/43011
	$(2.0, 10^{-5})$	$(0.1, 0.48)$	500/43011
	$(4.0, 10^{-5})$	$(0.1, 0.25)$	500/43011
	$(8.0, 10^{-5})$	$(0.1, 0.15)$	500/43011
	$(16.0, 10^{-5})$	$(0.1, 0.11)$	500/43011
Compas	$(2.0, 10^{-4})$	$(0.1, 0.9)$	100/3694
	$(4.0, 10^{-4})$	$(0.1, 0.48)$	100/3694
	$(8.0, 10^{-4})$	$(0.1, 0.27)$	100/3694
	$(16.0, 10^{-4})$	$(0.1, 0.16)$	100/3694
	$(32.0, 10^{-4})$	$(0.1, 0.11)$	100/3694

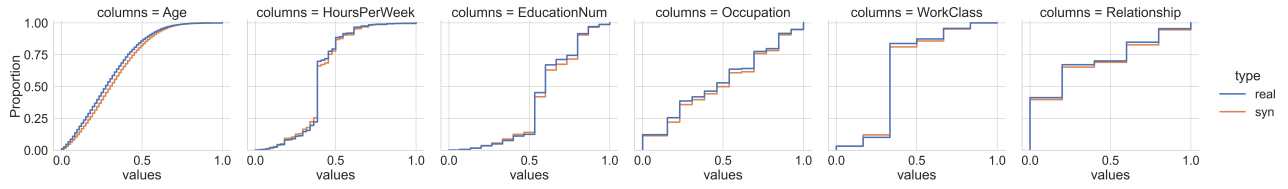
## C MITIGATION

### C.1 DP-WGAN

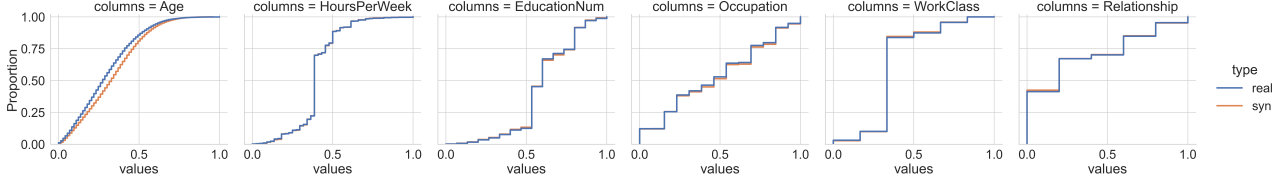
We show the ECDFs of marginals for  $(\epsilon, \delta)$ -DP synthesized data in Fig. 18. Smaller training data usually gains less satisfactory generation quality under DP training with a similar privacy budget.

### C.2 Naive and Improved Defenses

We show additional ECDFs of marginals for “Remove Colliding Members” mitigation and “GAN-constrained Training” mitigation in Figs. 19, 20, and 21.

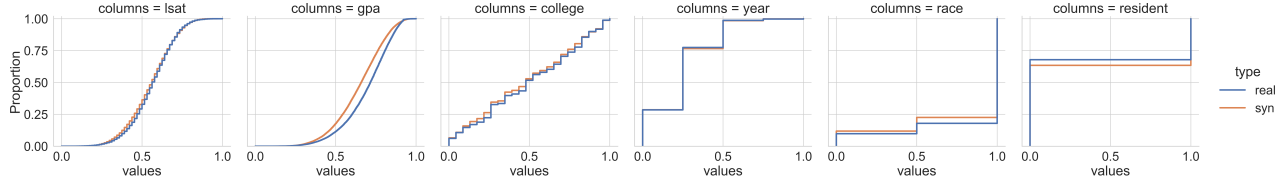


(a) Adult ECDF, Remove Overlapping

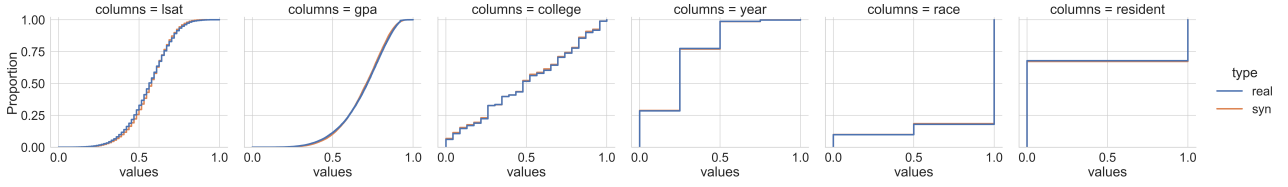


(b) Adult ECDF, GAN-constrained Training

**Figure 19: ECDF comparisons for “Remove Overlapping” mitigation and “GAN-constrained Training” mitigation.**

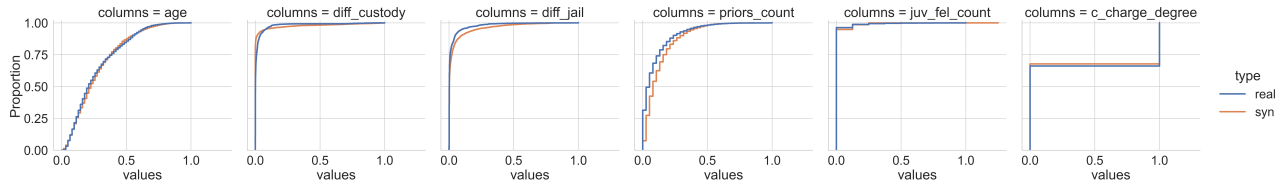


(a) Lawschool ECDF, Remove Overlapping

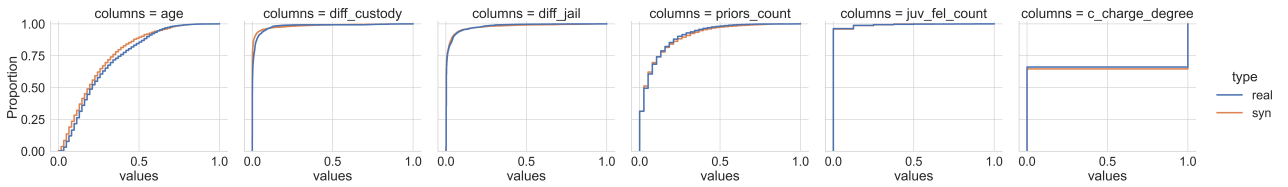


(b) Lawschool ECDF, GAN-constrained Training

**Figure 20: ECDF comparisons for “Remove Overlapping” mitigation and “GAN-constrained Training” mitigation.**



(a) Compas ECDF, Remove Overlapping



(b) Compas ECDF, GAN-constrained Training

**Figure 21: ECDF comparisons for “Remove Overlapping” mitigation and “GAN-constrained Training” mitigation.**