Differentially Private Data Generation with Missing Data

Shubhankar Mohapatra, Jianqiao Zong, Florian Kerschbaum, Xi He
University of Waterloo
shubhankar.mohapatra,jianqiao.zong,florian.kerschbaum,xi.he@uwaterloo.ca

ABSTRACT

Despite several works that succeed in generating synthetic data with differential privacy (DP) guarantees, they are inadequate for generating high-quality synthetic data when the input data has missing values. In this work, we formalize the problems of DP synthetic data with missing values and propose three effective adaptive strategies that significantly improve the utility of the synthetic data on four real-world datasets with different types and levels of missing data and privacy requirements. We also identify the relationship between privacy impact for the complete ground truth data and incomplete data for these DP synthetic data generation algorithms. We model the missing mechanisms as a sampling process to obtain tighter upper bounds for the privacy guarantees to the ground truth data. Overall, this study contributes to a better understanding of the challenges and opportunities for using private synthetic data generation algorithms in the presence of missing data.

1 INTRODUCTION

Our world as we see it today revolves around private data regarding our medical, financial, and social information. It is sometimes imperative to query such data for research and advancement of science [8, 46]. Many industries also use statistics from private data to improve their products and user experience. However, reckless data sharing for data-driven applications and research causes great privacy concerns [11, 41] and penalties [1]. As a response, differential privacy (DP) [26] has emerged as a standard data privacy guarantee which has now been adopted by government agencies [5, 39] and companies [30, 37, 44]. Informally, DP guarantees that the output distribution of an algorithm is similar with or without a particular individual in the dataset. A privacy budget is set to limit the total privacy loss and each query (e.g., releasing statistics [5, 17], building prediction models [3, 70], and answering SQL queries [34, 44, 48, 60]) consumes part of the privacy budget, and once that budget is used up, no more queries can be answered directly. An alternative way is to generate a synthetic dataset using the privacy budget. The synthetic dataset, once generated, can be made public, and the analyst can use it for any number of downstream tasks [13, 21, 22, 45, 97].

Despite a number of work [35, 45, 73, 97] that succeed in generating synthetic data with DP guarantees, they only look at a simple scenario where the input data has no missing values. Several prior studies [36, 56] have reported on the prevalence of missing data in various fields. For instance, a study of 9 publicly available healthcare datasets commonly used in machine learning research found that the proportion of missing values ranged from 0.2% to 78.6% [36]. The presence of missing data can be attributed to multiple reasons, such as human errors [7] and privacy regulations like GDPR [1] which allow people the "right to forget" where one may ask their data to be deleted completely [75]. In our work, we ask how missing data will affect the quality of the synthetic data generated by mechanisms that offer DP guarantees. Our preliminary study shows that existing DP mechanisms have 4%-18.5% decay in the F1-score of downstream ML tasks on the synthetic dataset generated from a dataset with 10% missing values as compared to that when generated from the complete dataset. The decay varies depending on the types of missing mechanisms and the types of data generation processes.

In our work, we formally define the research problem of generating synthetic data for sensitive data with missing values using DP. We consider a missing mechanism that takes complete ground truth data and outputs data with missing values. Under this setup, we can offer DP to either *the incomplete data* or *the ground truth data*. For each privacy guarantee, we study how to handle missing data in the synthetic data generation process. While it is desired to have a unified comprehensive approach to solve the problem, we identify multiple challenges in achieving such a solution. First, there is no straightforward winning algorithm for DP synthetic data generation even without missing data as demonstrated by prior benchmarking work [83]. Second, the design space for a solution is huge for dealing with missing data. We explore several techniques, including the vanilla approach that uses complete rows only [66], common imputation techniques (e.g., statistical methods, machine learning methods) [50, 56, 67, 78], and other differentially private imputation methods [24, 35]. We show that all these methods have their limitations, such as discarding too many rows or incurring high privacy costs, leading to poor-quality synthetic data. Therefore in our work we present a comprehensive list of possible approaches to solve the problem including two vanilla approaches and three adaptive approaches, as a contribution to understanding the design space for this new problem.

In addition, we show the relationship between the privacy guarantee for these DP mechanisms that they offer for the incomplete data and that for the ground truth data. To do so, we model the missing mechanism as a sampling process and obtain a tighter upper bound for the privacy loss to the ground truth data via sampling amplification techniques [12, 81]. Unlike prior work for sampling amplification that considers a random subset, we make use of the randomness due to missing values to amplify the privacy for ground truth data. The major contributions of our work are as follows:

- We are the first to formalize and study the problems of DP synthetic data with missing data. Our results show that existing algorithms have a decrease of 5-23% in utility with $\leq 5\%$ missing values and a decrease of 10-190% with $\leq 20\%$ missing values.
- We develop three novel adaptive approaches, each tailored to an existing category of DP mechanisms, that seamlessly combine dealing with missing data along with the learning process. Our evaluation shows that they improve the utility of the synthetic datasets by up to 15-72%. These simple yet effective approaches

sometimes even achieve the same utility as the synthetic data generated from the no-missing ground truth data.

- We differentiate the privacy guarantees for both incomplete and ground truth data. Our analysis develops sufficient conditions that when satisfied, the algorithms for the incomplete data can be used to achieve privacy for the ground truth data.
- We are the first to apply amplification due to missing mechanisms and tighten the privacy bound for ground truth data. The amplified ground truth privacy is 0.1-0.65x the privacy achieved for the incomplete data with 10-50% missing values.

2 PRELIMINARIES

We consider a database relation $R = \{A_1, \cdots, A_k\}$ with k attributes, and a database instance D consisting of n rows. We use D_i to refer to the ith row of D, and D_{ij} to refer to the jth attribute of row D_i. We also use $S_{:i}$ to denote all elements from 1 to i in a sequence S.

2.1 Missing Data

For missing data, we define a missingness indicator matrix $M = [\ldots, m_{ij}, \ldots]$ of size $n \times k$, where $i \in [1, n]$, $j \in [1, k]$ and, shorthand m_i to point to i^{th} row of M. Each cell of M has one-to-one relation with D such that, $m_{ij} = 1$ if D_{ij} is missing and $m_{ij} = 0$ otherwise.

Missing data is classified into different types using missing mechanisms. A missing mechanism $M_\phi : \mathcal{D} \rightarrow \mathcal{D}$ takes as input the ground truth dataset $\bar{D}$ and outputs an incomplete dataset D. It is parameterized by Φ, a set of probabilities, which refers to the set of probabilities that control the unknown missing data process. Three missing types can be defined using Φ and the conditional distribution of missing indicator m_i given the dataset D_i [56, 78].

Missing completely at random (MCAR) assumes the probability of missingness is completely independent of the data. Under MCAR, any two rows of the dataset, regardless of their values, for the same attribute have the same probability of having a missing value. Hence, the parameter set Φ consists of $\{\phi_j | j \in [1, k]\}$, where ϕ_j is the probability of any row having a missing value for the jth column. For $j \in [1, k]$ and $i \in [1, n]$, $\Pr[m_{ij}|D] = \Pr[m_{ij}] = \phi_j$. Hence, the probability of a row having no missing values is $\prod_{j=1}^{k}(1 - \phi_j)$.

Missing at random (MAR) captures the scenario when the probability of missingness is independent of the missing values given the observed data. In other words, under MAR how likely a value is to be missing can be estimated based on the non-missing data. Consider examples, 1) Young people have missing IQ (because they haven't taken an IQ test yet), and MAR models the same probability of missing IQ attribute for rows of the same age, regardless of their IQ values; 2) Businessmen are less likely to share their income, and MAR models the same probability for income values for rows that have an occupation as 'Business.' MAR, therefore, is parameterized by a set of conditional probabilities Φ where each $\phi \in |\Phi|$ maps relationships between observed and missing values in the dataset. For $\phi_x \in \Phi$ that models the jth column's missingness, and for $i \in [1, n]$, $\Pr[m_{ij}|D_{(0)}] = \Pr[m_{ij}|D_{(0)}, D_{ij}] = \phi_x$, where $D_{(0)}$ refers to the observed attributes in the dataset.

Missing not at random (MNAR) captures the scenario when given all the observed information, the probability of missingness depends on any other unobserved missing values in the dataset. Consider examples, 1) Students with missing attendance also have missing scores, and the probability of the missing scores depending on missing attendance 2) People who smoke don't want to mention they smoke. Here MNAR models the probability of missing smokers based on the attribute of smoking. With MNAR missingness, Φ consists of a set of conditional probabilities that map the probability of an attribute to be missing given its own value.

Datasets often contain various missing data types, but identifying them without domain knowledge remains challenging [54, 55]. While statistical tests exist for MCAR cases (e.g., Little's MCAR test, pattern mixture models) [54, 55], identifying MAR and MNAR remains unsolved due to complex interactions between observed and unobserved variables [87]. Despite binary indicator tests based on modeling for MAR, these lack conclusiveness and are sensitive to analysis assumptions. For private datasets, these tests also need to be computed privately by using a portion of the privacy budget.

2.2 Differential Privacy

Differential privacy (DP) [27, 28] is used as our measure of privacy.

Definition 2.1 (Differential Privacy (DP) [27, 29]). A randomized algorithm M achieves (ϵ, δ)-DP if for all $Z \subseteq \text{Range}(\mathcal{M})$ and for two neighboring databases $D, D' \in \mathcal{D}$ that differ in one row:

$$\Pr[\mathcal{M}(D) \in Z] \leq e^{\epsilon} \Pr[\mathcal{M}(D') \in Z] + \delta.$$

The privacy cost is measured by the parameters (ϵ, δ), often referred to also as the privacy budget. The smaller the privacy parameters, the stronger the offered privacy.

Gaussian mechanism [29] and Laplace mechanism [28] are two such widely used DP algorithms. Given a function $f : \mathcal{D} \rightarrow \mathbb{R}^d$, the Gaussian mechanism adds noise sampled from a Gaussian distribution $\mathcal{N}(0, S_f^2\sigma^2)$ to each component of the query output, where σ is the noise scale and S_f is the L_2 sensitivity of function f, which is defined as $S_f = \max_{D,D' \text{differ in a row}} ||f(D) - f(D')||_2$. For $\epsilon \in (0, 1)$, if $\sigma \geq \sqrt{2\ln(1.25/\delta)}/\epsilon$, the Gaussian mechanism satisfies (ϵ, δ)-DP. Laplace mechanism works similarly but with the noise from the Laplace distribution and the L_1 sensitivity. Both these mechanisms have been applied to answer counting queries [51] and is widely used in estimating low dimensional statistics about the dataset.

Complex DP algorithms can be built from these basic algorithms following two important properties of DP: 1) Post-processing [27] states that for any function g defined over the output of the mechanism $\mathcal{M}$, if $\mathcal{M}$ satisfies (ϵ, δ)-DP, so does $g(\mathcal{M})$; 2) Composability [26] states that if $\mathcal{M}_1, \mathcal{M}_2, \cdots, \mathcal{M}_k$ satisfy (ϵ_1, δ_1)-, (ϵ_1, δ_1)-, $\cdots$, (ϵ_k, δ_k)-DP, then sequentially applying these mechanisms satisfies $(\sum_{i=1}^{k} \epsilon_i, \sum_{i=1}^{k} \delta_i)$-DP.

2.3 DP Synthetic Data Generation

A common DP study is to generate synthetic data given a fixed privacy budget. The synthetic data, once generated, can be made public and all queries on this dataset come for free due to the post-processing property of DP. There are three main approaches for DP synthetic data generation [83]:

Statistical approaches rely on estimating low-dimensional statistics about the dataset such as marginals [58, 76, 91]. These approaches can be made better by finding the correlation between

attributes. Techniques for improvement include probabilistic models [47], Bayesian models [52, 74, 97] and undirected graphs [22, 59]. Statistical approaches capture the underlying distribution of the correlated independent attributes very well but fail to imbibe complex relationships between multiple attributes.

Deep learning approaches are promising for generating synthetic data [19, 38, 79], particularly with autoencoders and generative adversarial networks (GAN). Autoencoders map data into a low-dimensional feature space and sample synthetic data from the low-dimensional space. GANs utilize a generator to produce fake examples and a discriminator to distinguish real from fake. DPSGD [3, 14, 80, 89] is commonly used for privacy. Various private approaches for autoencoders [4, 6, 73] and GANs [33, 45, 86, 92] have been proposed, effective on image data but challenged with tabular data due to poor encoding. Conditional GANs [93] and private versions [85] address this by sampling based on conditional probabilities of categorical attributes.

Mixed approaches are inspired by both the above approaches and try to preserve both low-dimensional statistics and high-level information. Some techniques include leveraging the dimensionality reduction via random orthonormal (RON) projection, the Gaussian generative model [18], combining denial constraints and attribute-wise embedding models [35] and Gretel.ai statistics [68].

3 PROBLEM STATEMENT

Consider a private dataset behind a privacy firewall with n rows and k attributes. A trusted curator aims to generate a synthetic version of the same size with an end-to-end (ϵ, δ)-DP guarantee while preserving maximum utility. Real-world data collected by curators may contain missing values, a scenario overlooked in prior work. We address this by formalizing two versions of the problem based on privacy considerations. First, we offer a DP guarantee for the incomplete dataset held by the data curator.

Problem 1. *[Privacy for Incomplete Data] Consider collecting data from a ground truth data $\bar{D}$ of n rows owned by n individuals, a missing mechanism $M_\Phi : \mathcal{D} \to \mathcal{D}$ is involved that takes in $\bar{D}$ and outputs a dataset D of n rows but with missing values. A trusted data curator uses this dataset D as input and aims to generate a synthetic data D^* of n rows with a mechanism $M : \mathcal{D} \to \mathcal{D}$ such that D^* minimizes $d(f(D) - f(D^*))$, where $f : \mathcal{D} \to \mathbb{R}^l$ is any utility metric function that the user is interested in, $d(\cdot, \cdot)$ is a distance metric and M offers (ϵ, δ)-DP to the input data D.*

In Section 4, we delineate multiple options for the data curator to deal with incomplete data, put forward challenges that come with them, and discuss which option might be the best and when. The first option is a naïve adaptation of prior work for DP data generation by simply discarding the rows with missing values [78]. We refer to this approach as *complete row only*. This approach can fail in many cases, as detailed in Section 4.1. For instance, if all rows have some missing values, then there will be no input data for the data generation methods. A second approach is to impute the missing parts with inferred values from the observed data. We denote this approach as *imputation first approach*. However, as our data is private, the imputation process needs to be privatized as well and the additional incurred privacy cost must be accounted for in the privacy budget. In section 4.1, we explore the privacy costs of imputation and show how they can be expensive in practice. Rather than having separate processes for imputation and synthetic data generation, we can integrate these two processes into one. This line of thought motivates us to a new approach, which we call *adaptive recourse approach*. In Section 4.2, we improve upon three categories of DP generation approaches and demonstrate their effectiveness in generating synthetic data from incomplete data. These strategies use no extra privacy budget and solely improve by observing available data in the dataset.

The incomplete data D can be modeled as a sample generated from a complete ground truth dataset $\bar{D}$ via a missing mechanism M_Φ. If the privacy goal is to protect the ground truth dataset $\bar{D}$ with DP guarantee, how will the problem and the solution be different? We formalize the second problem as follows.

Problem 2. *[Privacy for Ground Truth Data] Consider the same setup as Problem 1. The trusted data curator uses the incomplete dataset D as input and aims to generate a synthetic data D^* of n rows with a mechanism $M : \mathcal{D} \to \mathcal{D}$ such that minimizes $d(f(\bar{D}) - f(D^*))$, where $f : \mathcal{D} \to \mathbb{R}^l$ is any utility metric function that the user is interested in, $d(\cdot, \cdot)$ is a distance metric and $M \circ M_\Phi$ offers $(\bar{\epsilon}, \bar{\delta})$-DP to the ground truth data $\bar{D}$.*

The problem mentioned above differs from our initial one only in the final aspect: instead of aiming for DP for the observed incomplete data, we target DP for the ground truth data. The missing mechanism limits the information available for synthetic data generation. Although initially appearing similar, ensuring privacy for incomplete data doesn't necessarily guarantee privacy for ground truth data. In Section 5, we delve into their relationship, showcasing scenarios where privacy for incomplete data may or may not extend to ground truth data. We also explore how the missing mechanism can serve as a sampling mechanism to enhance privacy and improve the utility of synthetic data.

4 PRIVACY FOR INCOMPLETE DATA

This section examines Problem 1 and explores methods for generating synthetic data from an incomplete private dataset. The section starts by discussing two vanilla methods that are found to be ineffective in the DP context and instead recommends adaptive recourse methods that are novel solutions that address both issues and produce better-quality synthetic data.

4.1 Vanilla Approaches

Complete row only and imputation first are two traditional methods for handling missing data. These methods either involve discarding rows with missing values or filling up missing information with values inferred from the observed data.

Complete Row Only Approach. This approach is effective when the missingness is completely at random (MCAR) since the distribution of each attribute remains the same after removing missing rows. However, for other types of missingness, such as missing at random (MAR) and missing not at random (MNAR), the complete row only approach can lead to biased results. Hence, a standard synthetic data generation algorithm that learns directly from the

remaining complete rows will result in a biased data distribution that is different from the ground truth data.

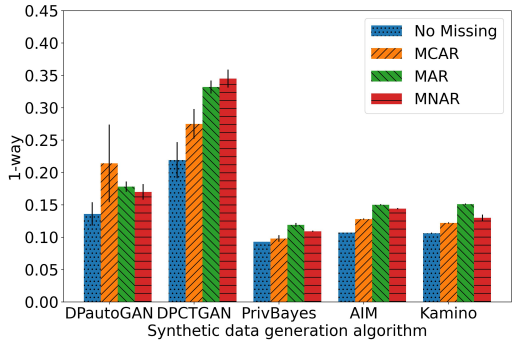


Figure 1: Complete row only approach results in poor results for MAR and MNAR missing mechanism.

Figure 1 illustrates the performance of five different DP synthetic data generation algorithms using the complete row only approach for the Adult dataset with various missing mechanisms. The results indicate that the complete row only approach performs poorly for missing at random (MAR) and missing not at random (MNAR) mechanisms, which introduce bias to the estimated distribution of attributes. This bias can directly affect statistical approaches such as PrivBayes which rely heavily on empirical estimation of marginals. The 1-way distance between the generated synthetic dataset and the original dataset confirms this observation.

Besides the potential bias issue for the complete row only approach, the number of complete rows remaining can be very small. For synthetic data generation methods involving large deep learning models such as GAN, feeding the training process with a small number of complete rows will result in a poor data generation model. This is because the learning process does not converge or/and the noise added for achieving DP overshadows the signals of the training samples. For example, the ground truth Adult dataset which has 32k rows reduces to $\approx$ 5k complete rows with 20% MAR and $\approx$ 1k complete rows with 20% MCAR/MNAR missing mechanism respectively. Our results show that the number of complete rows plays a vital role in the performance of the synthetic data generation algorithms. We discuss this in detail in Section 6 where we study several prior work approaches and evaluate them on the different missing data scenarios.

Imputation First Approach. Imputation is vastly used in practice where the missing data are filled up with values inferred from the observed data. There are multiple ways to impute missing values in the dataset, including:

(1) Statistical methods: Each attribute of the dataset is modeled separately using statistical methods such as mean, median, and mode. The model is then used to fill up the missing values of the attribute [56, 78]. For example, in Figure 2, for the left tables, we use the median of the observed values of column A, to fill up the missing value of the 4th row.
(2) Hot and cold deck methods: This imputation technique replaces every missing value with another value from the same dataset (Hot deck) or from a proxy dataset (Cold deck) [9, 67]. The missing cells of the incomplete row are then filled up from the closest similar row. Similarity metric like cosine distance or ℓ_2 distance can be used.
(3) ML imputation: Machine learning (ML) based approaches are common for missing data imputation [43, 50]. An ML model is trained to predict the missing values of an attribute based on other non-missing attributes in the dataset as training features.

In our private dataset setting, imputations must also be conducted privately. A straightforward yet ineffective method involves randomly selecting values from the missing attribute's domain, as illustrated later (Figure 5). We skip analysis of the cold deck imputation as finding another similar dataset is impractical for private datasets.

DP imputation can be approached in two manners. The first involves splitting the privacy budget, allocating a portion for imputation and the remainder for synthetic data generation. However, this approach is challenging due to budget allocation and choice of imputation algorithm. Additionally, some imputation techniques such as the hot deck imputation that are row specific (replicates the missing value in a row based on some other observed value of a different user) cannot be performed in the DP setting. Randomizing this row to achieve DP introduces too many errors to the dataset.

The second way is to formulate imputation as a transformation of the dataset and calculate the associated privacy cost as an end-to-end algorithm. We use the notion of stability (Theorem 4.1) to calculate the privacy costs of these transformations.

Theorem 4.1. *[60] We say a transformation $T(\cdot)$ is c-stable, if the distance between $T(D)$ and $T(D')$ is at most c times the distance between D and D'. The composite mechanism $\mathcal{M} \circ T$ then becomes $(c \cdot \epsilon, \delta)$-DP, for any mechanism $\mathcal{M}$ which is (ϵ, δ)-DP.*

Lemma 4.2. *Consider a transformation $T_A(\cdot)$ for imputing attribute A, which takes in the incomplete dataset D as part of the input and outputs a dataset D' with the complete values for attribute A. Then, the stability of $T_A(\cdot)$ is $c = m_A + 1$ where m_A refers to the number of missing values for the attribute A.*

Proof. As the neighboring databases D and D' differ by a row, $T_A(\cdot)$ uses two different values x and x' to impute the missing values in D and D' respectively. As there are $m_{A(1)}$ rows in both D and D' that have missing values for attribute A, the resulted imputed databases, $T(D)$ and $T(D')$ have $m_{A(1)} + 1$ number of rows (include the row that D and D' differ). □

Using the above Lemma we can see that applying a sequence of imputation functions over the attributes of a dataset $(\cdots, T_{A_i}, \cdots)$, the difference in the resulted datasets can be very large when the input dataset differs in a single row. Note that these results hold even if imputation functions for two attributes are not the same.

Theorem 4.3. *The composite mechanism $\mathcal{M} \circ T$ on a dataset D with n rows is $n\epsilon$-DP, where $\mathcal{M}$ is a ϵ-DP mechanism, and T is a sequence of imputation functions performed to each attribute of D.*

Proof. Consider a worst-case scenario: D and D' differ in a single row that does not have any missing values, and the rest of the rows have only one attribute with missing values, i.e., $\sum_i m_{A_i} = n-1$. As T uses the complete row to impute all the missing values, all rows will be affected and the overall cost of $M \circ T$ will be $n\epsilon$-DP. □

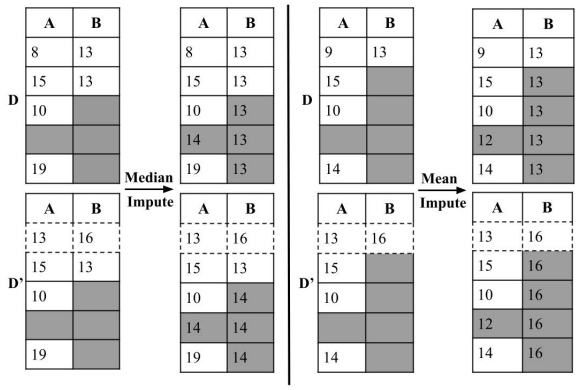


Figure 2: Illustration of worst case statistical imputations.

Example 4.4. In Figure 2, we illustrate two worst-case toy examples for a dataset with two columns using mean and median imputations. The gray color indicates missing values and dotted lines denote differing rows between the top and the bottom datasets for each example. The missing values in both columns in the left and right examples are filled up with mean and median functions respectively. After applying imputations on the neighboring datasets D and D', the number of rows in column B in both examples starts to differ by 4 rows and 5 respectively including the imputed values and the differing row. In such a scenario, one needs to pay 4ϵ or 5ϵ privacy cost to ensure DP to the incomplete data.

4.2 Adaptive Recourse Approach

Both aforementioned approaches suffer from inefficiencies in data or privacy budget utilization. The complete row approach discards partial rows, wasting potential learning data, while the imputation-first method incurs high privacy costs. These challenges prompt us to modify existing synthetic data generation techniques to optimize both dataset information and privacy budget usage, which we term *adaptive recourse*. The concept involves employing the privacy-preserving learning aspect of synthetic data generation for both imputation and synthetic data sampling. This offers two advantages: the privacy budget is dedicated solely to learning a single model, and the imputation process generates more comprehensive training examples, enhancing model utility. We have selected three representative DP data generation models—generative adversarial networks (GAN), partial marginal observation, and column-wise data generation—as proof-of-concept for these adaptive approaches, which can extend to other existing or new DP models.

GAN-based adaptive recourse. In non-private literature, several approaches use the GAN framework to deal with missing data [53, 57, 95, 96]. We choose to privatize an approach called MisGAN [53], which allows us to simultaneously demonstrate the power of learning the data distribution and the missingness pattern for GAN-based algorithms. We call its DP version, DP-MisGAN, as shown in Algorithm 1. We first describe the high-level architecture of MisGAN and DP-MisGAN, and then we highlight the enhancement in this approach.

Algorithm Overview: MisGAN/DP-MisGAN trains two generator-discriminator pairs — one for learning the data distribution and the other for learning the missingness pattern. The training spans E epochs, with each epoch sampling $|D|/B$ sized subsets from the dataset D without replacement (Line 4). Each subset S_t is processed with real data x_{data} (Line 6) and its corresponding missing mask x_{mask}(Line 7). The missing mask x_{mask}, computed from the missing indicator matrix M, marks where data is missing as 1 and 0 otherwise. Missing values in real data are replaced with 0s (Line 8). Two fake examples y_{data} and y_{mask} are generated by passing random Gaussian noise through data and mask generators (Line 9). These generators are updated using gradient descent, with discriminators learning true distributions in one phase (Line 10) and generators updating within specified T_G intervals in the second phase (Line 11). In each generator interval, two fake samples y_{data} and y_{mask} are again generated similarly from the two generators (Line 12) and gradients from the discriminator is computed (Line 13). Note that the non-private MisGAN uses these gradients directly to update the parameters of the two generators (Line 16) and releases both generators in the end. Finally, after the training is completed for E epochs, the discriminators are thrown away, and the privately learned data generator is used to sample synthetic data (Line 20) .

Highlights: In our DP-MisGAN, we clip and then noise the gradients learned for the generators using the *sampled Gaussian mechanism* (SGM) [62] to ensure privacy (Lines 14-15). This step is different from prior efforts for DPCTGAN [32] and DPautoGAN [82], which privatize non-private optimizer of the discriminators using DPSGD [3, 14, 80, 89]. This divergence in technique leads to notable difference in the algorithm's utility: while the discriminators become noisy due to the introduction of noise, the generator's gradient calculation, which relies on discriminator weights (Line 13), is also affected. We observe that we only need to publish the generators, not the discriminators. Hence, we resort to the gradient sanitization (GS) approach [20] to perturb only the gradients of the generators (Lines 14-15), without affecting the utility of the discriminator. In addition, the GS approach also allows us to skip the hyperparameter tuning for the gradient clip C, which can be vastly detrimental if set wrong [63]. Unlike the standard gradient clipping to bound the sensitivity of the gradient norm by C, i.e., $g/max(1, \|g\|_2/C)$, we clip the gradient by $g/max(1, \|g\|_2)$ (Line 15) by considering a Wasserstein-Gan (WGAN) [10] framework with an additional gradient penalty term in the loss function of the discriminator that enforces the ℓ_2-norm of the discriminator gradients to be naturally close to 1. A tight bound for the privacy loss of the training procedure can be calculated using Rényi-DP (RDP).

Definition 4.5 (Rényi-DP [61]). A randomized algorithm M with domain $\mathcal{D}$ is (α, ϵ)-RDP at order $\alpha > 1$, for any pair of neighbouring databases $D, D' \in \mathcal{D}$ that differ in one tuple. Let P_D and $P_{D'}$ be the output probability density of $\mathcal{M}(D)$ and $\mathcal{M}(D')$ respectively. Then, it holds that: $\frac{1}{\alpha-1} \log \mathbb{E}_{x\sim\mathcal{M}(D')} \left(\frac{P_D(x)}{P_{D'}(x)}\right)^{\alpha} \leq \epsilon$.

The post-processing and composability properties of DP also apply to RDP. Specifically, if a sequence of adaptive mechanisms $\mathcal{M}_1$, $\mathcal{M}_2$, $\cdots$, $\mathcal{M}_k$ satisfy (α, ϵ_1)-, (α, ϵ_2)-, $\cdots$, (α, ϵ_k)-RDP, then the composite privacy loss is $(\alpha, \sum_{i=1}^{k} \epsilon_i)$-RDP. The RDP privacy

Algorithm 1 DP-MISGAN

Require: Incomplete dataset D, noise scale σ, epochs E, learning rates η_D and η_G, generator interval T_G, batch size B, missing indicator matrix M

Initialize data generator θ_G^D and discriminator θ_D^D
Initialize mask generator θ_G^M and discriminator θ_D^M
for i in $[1, \ldots, E]$ **do**
  Subsample dataset D into $\{S_k\}_{k=1}^{k=(|D|/B)}$ subsets
  for t in $[1, \ldots, |D|/B]$ **do**
    Set real data $x_{data} = S_t$
    Sample real mask x_{mask} from missing indicator $M(S_t)$
    Fill missing values in x_{data} with 0
    Generate fake data and mask
    $y_{data} = \theta_G^D(z)$ where $z \sim \mathcal{N}(0, 1)$
    $y_{mask} = \theta_G^M(z)$ where $z \sim \mathcal{N}(0, 1)$
    Update $\theta_D^D = \theta_D^D - \eta_D \nabla_{\theta_D^D}[\frac{1}{B}(\theta_D^D; x_{data}) - \frac{1}{B}(\theta_D^D; y_{data})]$
    $\theta_D^M = \theta_D^M - \eta_D \nabla_{\theta_D^M}[\frac{1}{B}(\theta_D^M; x_{mask}) - \frac{1}{B}(\theta_D^M; y_{mask})]$
    if t in interval of T_G **then**
      Generate fake data and mask
      $y_{data} = \theta_G^D(z)$ where $z \sim \mathcal{N}(0, 1)$
      $y_{mask} = \theta_G^M(z)$ where $z \sim \mathcal{N}(0, 1)$
      Compute data and mask gradient
      $g_d = \nabla_{\theta_G^D}(\theta_D^D; y_{data})$, $g_m = \nabla_{\theta_G^M}(\theta_D^M; y_{mask})$
      Clip each gradient $\bar{g}_{id} = g_{id}/\max(1, \|g_{id}\|_2)$
      $\bar{g}_{im} = g_{im}/\max(1, \|g_m\|_2)$
      Compute noisy gradients $\tilde{g}_{id} = \bar{g}_{id} + 2\sigma\mathcal{N}(0, 1)$
      $\tilde{g}_{im} = \bar{g}_{im} + 2\sigma\mathcal{N}(0, 1)$
      Update generators $\theta_G^D = \theta_G^D - \eta_D \frac{1}{B}\tilde{g}_{id}$
      $\theta_G^M = \theta_G^M - \eta_D \frac{1}{B}\tilde{g}_{id}$
    end if
  end for
end for
Generate and return synthetic dataset from generator θ_G^D

loss of the *sampled Gaussian mechanism* (SGM) [62] is given using Lemma 4.6.

Lemma 4.6. *Given a database D and query $f : \mathcal{D} \to \mathbb{R}^d$ with sensitivity S_f, returning $f(\{x \in D \mid x \text{ is sampled with probability } r\}) + \mathcal{N}(0, S_f^2\sigma^2\mathbb{I}^d)$ results in the following RDP cost for an integer moment α.*

$$R_{\sigma,r}(\alpha) = \begin{cases} \frac{\alpha}{2(S_f\sigma)^2} & r = 1 \\ \sum_{k=0}^{\alpha} \binom{\alpha}{k}(1-r)^{\alpha-k} r^k \exp(\frac{\alpha^2-\alpha}{2(S_f\sigma)^2}) & 0 < r < 1 \end{cases}$$

We can now calculate the total RDP cost of DP-MisGAN by composing the cost of each SGM in the training procedure.

Theorem 4.7. *The total RDP cost of DP-MisGAN is*

$$R(\alpha) = 2\lceil\frac{T}{T_G}\rceil \sum_{k=0}^{\alpha} \binom{\alpha}{k}(1-\frac{B}{|D|})^{\alpha-k}\frac{B}{|D|}^k \exp(\frac{\alpha^2-\alpha}{8\sigma^2})$$

Proof. Let the gradient clipping procedure of the discriminator in DP-MisGAN be $f = g/\max(1, \|g\|_2/C)$. The sensitivity S_f can be thus be derived by the reversed triangle inequality.

$$S_f = \max_{D,D'} \|f(D) - f(D')\| \leq 2C \leq 2$$

The last inequality follows as we set $C = 1$ in DP-MisGAN. Each SGM in DP-MisGAN (Algorithm 1) has sampling probability of $\frac{B}{|D|}$ and thus the RDP cost of each SGM can be derived using Lemma 4.6 as $\sum_{k=0}^{\alpha}\binom{\alpha}{k}(1-\frac{B}{|D|})^{\alpha-k}\frac{B}{|D|}^k \exp(\frac{\alpha^2-\alpha}{8\sigma^2})$. DP-MisGAN consists of two generators that are updated every generator interval T_G each. Therefore, using the composition property of RDP, we can compose the costs for both the discriminators in DP-MisGAN as $R(\alpha) = 2\lceil\frac{T}{T_G}\rceil\sum_{k=0}^{\alpha}\binom{\alpha}{k}(1-\frac{B}{|D|})^{\alpha-k}\frac{B}{|D|}^k \exp(\frac{\alpha^2-\alpha}{8\sigma^2})$. □

Algorithm 2 PrivBayes Enhanced (PrivBayesE)

Require: Incomplete dataset D, Attributes $\mathcal{A}$, Privacy budget ϵ_1, ϵ_2

Initialize Bayesian network B of degree k and $V = \phi$
Sample X_1 from $\mathcal{A}$ and add (X_1, ϕ) to B; add X_1 to V
for $i = 2 \ldots |\mathcal{A}|$ **do**
  Initialize $\Omega = 0$
  For each $X \in \mathcal{A} \setminus V$ and each $\Psi \in \binom{V}{k}$; add (X, Ψ) to Ω
  Select a pair (X_i, Ψ_i) from Ω with maximal mutual information in complete rows for attributes X_i in D using exponential mechanism of budget $\epsilon_1/|\mathcal{A}|$
  Add (X_i, Ψ_i) to B; add X_i to V
end for
Initialize synthetic dataset D^*
for $i = 1 \ldots |\mathcal{A}|$ **do**
  Compute distribution from non-missing values $\Pr[X_i, \Psi_i]$ from the complete rows of X_i in D
  Learn $\Pr^*[X_i, \Psi_i]$ with Laplace mechanism at budget ϵ_2
  Set negative values to 0 and normalize
  Sample from $\Pr^*[X_i, \Psi_i]$ and add to D^*
end for
Return D^*

By the tail bound property of RDP [61], we can convert the RDP cost of DP-MisGAN to (ϵ, δ)-DP, where ϵ is computed by

$$\epsilon(\delta) = \min_{\alpha} R(\alpha) + \frac{\log(1/\delta)}{\alpha - 1}, \tag{1}$$

for a given δ. In practice, given the parameters $\epsilon, \delta, B, T, T_G$, the σ is calculated using Theorem 4.7 and the order α is usually searched within a range of user input values [88].

We expect DP-MisGAN to perform better than the naive GAN approaches because it learns from both the complete rows as well as the incomplete rows of the dataset. Furthermore, as DP-MisGAN learns the missing data pattern of the incomplete dataset, we anticipate that it will capture more information in complex MAR and MNAR missing mechanisms.

Flexibility: The framework of DP-MisGAN can be used for any GAN method. The core idea of the change is to encompass two discriminator-generator pairs for learning missing data and synthetic data generation. However, to achieve better privacy accounting, it is important to use the GS approach with a WGAN structure and discard the discriminator after training.

Partial marginal observation-based adaptive recourse. This approach can be applied to algorithms that use low dimensional marginal queries [47, 52, 58, 59, 74, 76, 91, 97]. Instead of discarding all the partially missing rows, only the rows with missing cells in the queried attributes are removed. Such a strategy is most helpful when only a subset of attributes have missing data. For example, with MAR missing mechanism, partial marginal observation can be learned from all the non-missing columns.

Algorithm overview: We extend PrivBayes [97] using this strategy and call it PrivBayes enhanced or PrivBayesE in short (Algorithm 2). Both PrivBayes and PrivBayesE learn Bayesian networks of degree k to know the correlated columns. The network B is initialized by adding the first attribute in the attribute list $\mathcal{A}$. The vertices that been discovered so far are stored in the list V (Lines 1-2). Next the algorithm loops over each attribute in $\mathcal{A}$ (Line 3) and generates $|V|$ choose k sets appended with every attribute seen so far $\mathcal{A} \setminus V$ and stores in a list Ω (Lines 4-5). The mutual information values for each pair in Ω is computed and the best one chosen using exponential mechanism using privacy budget $\epsilon_1/\mathcal{A}$ is added to the network B (Lines 6-7). PrivBayes/PrivBayesE then generate a synthetic dataset D^* using this network B in sequence (Line 10). Marginals are computed for each attribute X_i with its most correlated attributes Ψ_i using privacy budget $\epsilon/2$ and added to D^* (Line 12-14).

Highlights: In PrivBayesE, modifications are made to the dataset generation process. Each time a marginal query is made, PrivBayesE learns from all non-missing information of the attribute(s) (Line 11). This improves upon the complete row approach by discarding missing rows on a smaller set of attributes, rather than from the entire dataset. This is particularly advantageous for scenarios like missing completely at random (MCAR), where analyzing more data aids in better estimating the true distribution of marginals, and missing at random (MAR), where some marginals are completely available, allowing estimation based on complete data. The privacy analysis of PrivBayesE mirrors that of PrivBayes, as PrivBayesE does not introduce additional queries to the dataset.

Flexibility: This enhancement can be applied as a wrapper to any method that makes use of low-dimensional marginals to generate synthetic data. In our paper, we choose PrivBayes as our baseline as it is the most simple and fundamental marginal based approach.

Column-wise data generation-based adaptive recourse. This approach can be applied to any algorithm that uses column-wise intermediate models to learn the data distribution. In such algorithms, a sequence of attributes is decided, and starting with the second attribute in sequence, a model is learned to predict the current attribute using previously learned ones.

Algorithm overview: We extend Kamino [35] using this strategy and call in Kamino impute or KaminoI in short (Algorithm 3). Kamino/KaminoI starts with deciding a sequence of attributes based on a given denial constraints Ψ (Line 1). The distribution of the first attribute in the computed sequence is learnt using all the non-missing cells (Line 3). This computed distribution is noised (Line 4) and values are sampled to populate the synthetic dataset (Line 5). For each new attribute Y in the sequence, Kamino/KaminoI learns a private intermediate model which uses all previously visited attributes X to predict the new attribute Y (Lines 7-8). This intermediate model is used to generate the values for the attribute Y in the synthetic data given the sampled values for X (Line 10).

Algorithm 3 Kamino Impute (KaminoI)

Require: Incomplete dataset D, Attributes $\mathcal{A}$, Constraints Ψ, Privacy budget ϵ_1, ϵ_2

1: Build sequence S of attributes $\mathcal{A}$ using constraints Ψ
2: Initialize synthetic dataset D^*
3: Compute distribution of first attribute $H = \Pr[S_1]$ using all non-missing values
4: Generate DP H^* by adding Gaussian noise of budget ϵ_1
5: Sample from H^* to populate $D^*[S_1]$
6: **for** $i = 2 \ldots |\mathcal{A}|$ **do**
7: Load training features $X = S_{:j}$, and target label $Y = S_j$
8: Train model $M = \theta(X, Y)$ privately with budget $\frac{\epsilon_2}{|\mathcal{A}|-1}$
9: Impute missing values in dataset $D[S_j]$ using M
10: Predict synthetic values $\Omega = M(D^*[S_{:j}])$ and fill $D^*[S_j] = \Omega$
11: **end for**
12: Return D^*

Highlights: KaminoI includes an additional imputation step to impute values of attribute Y (Line 9), utilizing the same intermediate model that was trained to predict attribute Y given attribute set X. This enhancement ensures that no missing values are discarded; instead, they are used to train intermediate models. It is worth noting that the sequence S significantly influences KaminoI's imputation process. In Kamino, the sequence is generated considering input constraints Ψ. However, if an attribute with many missing values is positioned early in sequence S, its imputation may be less effective. To optimize imputation, attributes not in constraints Ψ can be ordered based on decreasing percentage of missing values. If available, clues from the missing mechanism can also help determine the sequence. For instance, with the MAR mechanism predicting missing IQ based on age, the age attribute can precede the age column in S. To ensure fair comparison, the same sequence as Kamino is used for KaminoI in the experimental section.

Flexibility: This enhancement is applicable to any method that iterates over the columns of the dataset. Such an algorithmic architecture allows for value imputation as learning progresses. Each time a model predicts the next attribute, it can also impute missing values using the same model, incurring no additional privacy costs. This strategy is particularly effective when missing data correlates with other attributes in the dataset, such as in the missing at random (MAR) scenario. While there's only one known private approach employing this strategy [35], several non-private approaches exist [77, 94]. Although PrivBayesE can use learned distributions to impute missing values for all visited attributes (Line 12 in Algoirthm 2), it only utilizes low-way marginals compared to KaminoI's larger models, which benefit more from imputation. Hence, we did not include this imputation step in PrivBayesE.

5 PRIVACY FOR GROUNDTRUTH DATA

In this section, we shift our focus to exploring the privacy implications for the ground truth data, which we approach as a distinct problem that closely relates to Problem 1. We first find that the solutions for Problem 1 do not always offer sufficient privacy for

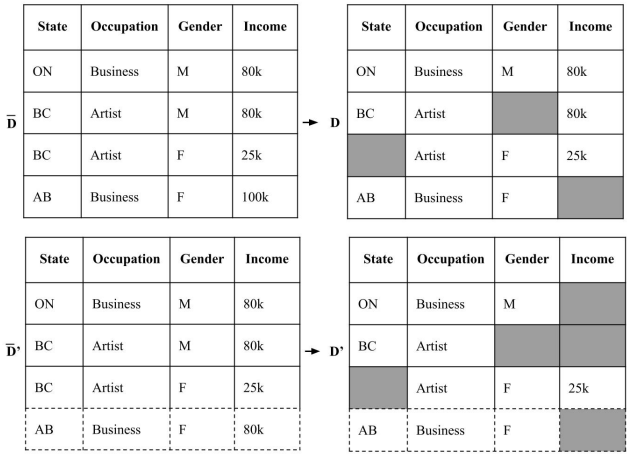


Figure 3: Example of private incomplete datasets generated from two neighbouring ground truth datasets. Gray denotes missing cells and dotted lines represent the differing row.

Problem 2, except when the probability of missing values in a row is independent of the other rows in the dataset (Section 5.1). Furthermore, we demonstrate that certain missing mechanisms, such as MCAR, allow a tighter privacy analysis for Problem 2 when applying the same solution from Problem 1 in Section 5.2.

5.1 Relationship to Problem 1

We have proposed multiple synthetic data generation algorithms M which train on the incomplete dataset D and achieve (ϵ, δ)-DP as solutions to problem 1. However, this incomplete dataset D results from a missing mechanism M_ϕ on the ground truth dataset $\bar{D}$. In problem 2, we study the same mechanisms M which train on D but discuss their privacy impact on $\bar{D}$. We do so by combining the missing mechanism M_ϕ and the synthetic data generation process M as a composite mechanism $M \circ M_\phi$.

It is important to note that just because M is a DP mechanism for incomplete data D, it does not necessarily mean that $M \circ M_\phi$ is DP for the ground truth data $\bar{D}$.

Example 5.1. In Figure 3, consider neighboring ground truth data $\bar{D}$ and $\bar{D}'$ differ in the last row's income value (100k v.s. 80k). Their income columns have an MNAR missing mechanism that hides the highest income value and their corresponding incomplete data D and D' then differ more than one row. This means that an ϵ-DP mechanism for incomplete data cannot guarantee the same level of privacy for the ground truth data.

The example above does not provide a strong privacy guarantee for the ground truth data because the probability of a row having missing values depends on the values of other rows. However, we can show that if M_ϕ enforces independent probabilities for each row to have missing values, a strong privacy guarantee applies to the ground truth data.

Theorem 5.2. *Let the missing mechanism M_Φ have independent randomness to hide the values of each row and $D = M_\phi(\bar{D})$. If M achieves (ϵ, δ)-DP for D, then $M \circ M_\Phi$ satisfies $(\bar{\epsilon}, \bar{\delta})$-DP for $\bar{D}$, where $\bar{\epsilon} \le \epsilon, \bar{\delta} \le \delta$.*

Proof. (sketch) As M_Φ has independent randomness to hide values of each row, given the ground truth data $\bar{D}$ and a possible incomplete dataset D, we have $\Pr[D|\bar{D}] = \prod_l \Pr[D_l|\bar{D}]$, where D_l refers to the value taken by the lth row. Consider neighboring groundtruth datasets $\bar{D}$ and $\bar{D}'$ differ in the ith row and any possible output O of $M \circ M_\Phi$. Let $\mathcal{D}$ be all possible incomplete datasets that can be outputted by M_Φ from $\bar{D}$ or $\bar{D}'$. We partition $\mathcal{D}$ into $\{\cdots, \mathcal{D}_j, \cdots\}$ such that all datasets with the same row values except the ith row are in the same group $\mathcal{D}_j$. Hence, for all $D \in \mathcal{D}_j$, they have the same probability for $\prod_{l \neq i} \Pr_{M_\Phi}[D_l|\bar{D}]$. Now we have

$$
\begin{aligned}
& \Pr[O|\bar{D}] \\
= & \sum_{\mathcal{D}_j} \sum_{D \in \mathcal{D}_j} \Pr_M[O|D] \Pr_{M_\Phi}[D|\bar{D}] \\
= & \sum_{\mathcal{D}_j} \sum_{D \in \mathcal{D}_j} (\Pr_M[O|D] \Pr_{M_\Phi}[D_i|\bar{D}] \cdot \prod_{l \neq i} \Pr_{M_\Phi}[D_l|\bar{D}]) \\
= & \sum_{\mathcal{D}_j} (\prod_{l \neq i} \Pr_{M_\Phi}[D_l|\bar{D}] \cdot \sum_{D \in \mathcal{D}_j} \Pr_M[O|D] \Pr_{M_\Phi}[D_i|\bar{D}]) \\
\le & \sum_{\mathcal{D}_j} (\prod_{l \neq i} \Pr_{M_\Phi}[D_l|\bar{D}'] \cdot \sum_{D' \in \mathcal{D}_j} (e^\epsilon \Pr_M[O|D'] + \delta) \Pr_{M_\Phi}[D'_i|\bar{D}']) \\
= & e^\epsilon \sum_{\mathcal{D}_j} \sum_{D' \in \mathcal{D}_j} \Pr_M[O|D'] \Pr_{M_\Phi}[D'|\bar{D}] + \delta \sum_{\mathcal{D}_j} \sum_{D' \in \mathcal{D}_j} \Pr_{M_\Phi}[D'|\bar{D}'] \\
= & e^\epsilon \sum_{\mathcal{D}_j} \sum_{D' \in \mathcal{D}_j} \Pr_M[O|D'] \Pr_{M_\Phi}[D'|\bar{D}] + \delta
\end{aligned}
$$

The inequality above is based on for any neighbors D and D', we have $\Pr_M[O|D] \le e^\epsilon \Pr_M[O|D'] + \delta$ and $\sum_{D \in \mathcal{D}_j} \Pr_{M_\Phi}[D_i|\bar{D}] = \sum_{D' \in \mathcal{D}_j} \Pr_{M_\Phi}[D'_i|\bar{D}'] = 1$. □

Theorem 5.2 says that the privacy bound for the ground truth dataset is lesser than equal to the bound of the incomplete dataset for a synthetic data generation algorithm if each row in the ground truth dataset has an independent probability of having missing values. Next we illustrate how to obtain a tighter privacy bound for the missing completely at random (MCAR) mechanism.

5.2 Privacy Amplification Due To MCAR

Missing completely at random (MCAR) enforces an independent probability of having missing rows for each attribute in the dataset. We use these probabilities to tighten the privacy bounds for ground truth data when the missing mechanism is MCAR. The technique we developed is inspired by the seminal work of privacy amplification due to sampling [12]. The premise of privacy amplification by subsampling is that we run a DP algorithm on some random subset of the data (e.g., sampled Gaussian mechanism, DP-SGD). The subset introduces additional uncertainty, which benefits privacy. Privacy amplification due to subsampling has been shown to work for many sampling methods (e.g., Poisson sampling, sampling with/without replacement) and for neighboring datasets which may differ with replacement or substitution. Privacy amplification by subsampling theorem 5.3 makes this intuition precise.

Theorem 5.3 (Sampling Amplification Theorem [12, 81]). *Consider an algorithm $M : \mathcal{D} \rightarrow \mathcal{D}$ that satisfies (ϵ, δ)-DP and a sampling mechanism $S(D)$ that samples a random subset U from dataset D of n samples. If $p = max_{i \in [n]} \Pr_U[i \in U]$, then the composite mechanism $M(S(D))$ offers (ϵ', δ')-DP where $\epsilon' = \log(1 + p(e^{\epsilon} - 1))$, $\delta' = p\delta$. For small values of ϵ, we have $\epsilon' = \log(1 + p(e^{\epsilon} - 1)) \approx p\epsilon$.*

In our missing data context, we note that for synthetic data generation algorithms that train on incomplete data, many rows are naturally discarded due to the presence of missing cells. We exploit this natural throwing out of rows as a sampling mechanism and show that it can be used to amplify privacy. Recall from Section 2.1 that MCAR enforces independent probability of having missing cells in the dataset for each attribute $\phi_1, \cdots, \phi_k$. We use these probabilities to propose our amplification results in Proposition 5.4.

Proposition 5.4. *Consider an MCAR mechanism $M_{\Phi} : \mathcal{D} \rightarrow \mathcal{D}$ with missing probabilities $\{\phi_1, \ldots, \phi_k\}$ over attributes $\{A_1, \ldots A_k\}$ of the input ground truth data $\bar{D}$ and outputs an incomplete dataset D. If an algorithm $M : \mathcal{D} \rightarrow \mathcal{D}$ takes in rows in D which have no missing values on attributes $\mathcal{A}_M \subseteq \{A_1, \ldots, A_k\}$, then $M \circ M_{\Phi}$ offers $(p\epsilon, p\delta)$-DP to the ground truth data $\bar{D}$ where $p_{\mathcal{A}_M} = \prod_{A_i \in \mathcal{A}_M}(1 - \phi_i)$. We call $\mathcal{A}_M$ an amplification attribute set for M and $p_{\mathcal{A}_M}$ the amplification factor of $\mathcal{A}_M$.*

Proof. A row in MCAR has $\prod_{i=1}^{i=l}(1 - \phi_i)$ probability of having no missing values and plugging in to Theorem 5.3. □

We note three important facts. First, if an algorithm M takes in rows with no missing values over an attribute set $\mathcal{A}_M$, then M also takes in rows with no missing values over an attribute set $\mathcal{A}'_M \subset \mathcal{A}_M$. In other words, if $\mathcal{A}_M$ is an amplification attribute set for M, then any subset of $\mathcal{A}_M$ is an amplification attribute set for M with amplification factor greater than that of $\mathcal{A}_M$. Second, when $\mathcal{A}_M = \emptyset$, $p_{\mathcal{A}_M} = 1$. Third and more importantly, as the dataset is read only once, each attribute can only be used once as an amplification factor. We can now use Proposition 5.4 and Theorem 5.3 in conjunction to show the privacy amplifications for the different algorithms we have discussed so far in our paper.

Use case 1: Privacy amplification for complete row only approach. Here we show how to apply Proposition 5.4 to all complete row only approaches (PrivBayes, Kamino, and GAN-based approaches). As these approaches take as input all attributes, the probability of seeing a row without missing values is $\prod_{i=1}^{i=k}(1 - \phi_i)$. The following example illustrates how this probability can be used to obtain a tighter privacy bound for the ground truth data.

Example 5.5 (MCAR amplification for complete row only approach). Consider the incomplete dataset from Figure 3. Let's assume that the missing data comes from an MCAR mechanism where the missing probabilities are $\phi_{State} = \frac{1}{4}, \phi_{Occupation} = 0, \phi_{Gender} = \frac{1}{4}, \phi_{Income} = \frac{1}{4}$. Given 4 DP sub-algorithms M_1, M_2, M_3, M_4 that each offer DP guarantee to the incomplete dataset D at budget $\frac{\epsilon}{4}$. M_1 computes the marginals of the complete rows over attribute <State>, M_2 over <Occupation>, M_3 over <Gender> and M_4 over <Gender, Income>. As all sub-algorithms take as input only the complete rows, using Proposition 5.4, the amplification is $\prod_i(1 - \phi_i) = 0.421$ and using Theorem 5.6 the final privacy is $\bar{\epsilon} = 4 * 0.421\frac{\epsilon}{4} = 0.421\epsilon$.

Use case 2: Privacy amplification for partial marginal observation approach. For partial marginal observation methods (e.g. PrivBayesE), calculating the amplification privacy cost is more complex. These methods involve multiple low-dimensional marginals with overlapping attributes. To determine the overall amplification for such algorithms, it is necessary to calculate the amplification for each marginal and carefully compose them. The complexity of this calculation arises from the optimal selection of the amplification attribute set for each marginal, which maximizes amplification while ensuring that each attribute is used only once. First, we consider a simple case that the amplification factors of all marginals are disjoint. In this scenario, we can compose the total privacy cost using Theorem 5.6 and demonstrate using Example 5.7.

Theorem 5.6. *Consider an MCAR mechanism M_{Φ}, and a sequence of j mechanisms $M_1, \ldots, M_j$ with DP guarantees of $\epsilon_1, \ldots, \epsilon_j$ to D and amplification attribute set $\mathcal{A}_{M_1}, \ldots, \mathcal{A}_{M_j}$ respectively. If their amplification attribute sets do not overlap, then these mechanisms offers DP to the ground truth data $\bar{D}$ at a cost of $\bar{\epsilon} = \sum_{i=1} p_{\mathcal{A}_{M_i}}\epsilon_i$.*

Proof. As all mechanisms M_i work on disjoint sets of attributes, their amplification attribute sets $\mathcal{A}_i$ are also disjoint. Furthermore as the missing probabilities are always ≤ 1, we always use all attributes in $\mathcal{A}_i$ amplify marginal M_i. We can then use Theorem 5.3 to calculate the final amplified privacy cost $\bar{\epsilon} = \sum_{i=1} p_{\mathcal{A}_{M_i}}\epsilon_i$. □

Example 5.7. Continuing from Example 5.5, assume we have the same dataset but use a partial observation algorithm. We consider only the sub-algorithms M_1, M_2, and M_4 for this example. The marginals for these sub-algorithms do not overlap and allow us to consider all engaging attributes as their amplification attribute set. Hence, by Theorem 5.6, M_1 is amplified using $p_{M_1} = 1 - \phi_{state} = \frac{3}{4}$, M_2 is amplified using $p_{M_2} = 1 - \phi_{occupation} = 1$ and M_4 using $p_{M_4} = (1 - \phi_{gender})(1 - \phi_{income}) = \frac{9}{16}$. The amplified privacy cost would thus be $\frac{3}{4}\frac{\epsilon}{3} + \frac{\epsilon}{3} + \frac{9}{16}\frac{\epsilon}{3} = 0.77\epsilon$.

The problem however gets more nuanced when two marginals have overlapping attributes. We show this in Example 5.8 by first showing a naïve composition and then an optimized one.

Example 5.8. Consider all 4 sub-algorithms in Example 5.5 and a partial observation algorithm. The marginals for sub-algorithms M_3 and M_4 overlap in the 'Gender' attribute with amplification factors $p_{M_3} = (1 - \phi_{Gender}) = \frac{3}{4}$ and $p_{M_4} = (1 - \phi_{Gender})(1 - \phi_{Income}) = \frac{9}{16}$ respectively. We cannot apply Theorem 5.6 on M_3 and M_4's entire attribute set as the corresponding amplification attribute sets would then overlap on the 'Gender' attribute. A naïve solution would be to amplify the DP mechanism with the most amplification and skip the others. In our example, we would amplify only M_4 with amplification of $\frac{9}{16}$ and skip amplification for M_3. The total amplified privacy cost would thus be $\bar{\epsilon} = \frac{3}{4}\frac{\epsilon}{4} + \frac{\epsilon}{4} + \frac{\epsilon}{4} + \frac{9}{16}\frac{\epsilon}{4} = 0.83\epsilon$. However, a better bound can be calculated if overlapping mechanisms were grouped together and amplified using the intersecting attribute. For instance, both M_3 and M_4 can be amplified by an amplification factor of $\frac{3}{4}$ using the amplification attribute set 'Gender', resulting in a total privacy loss of $\frac{3}{4}\frac{\epsilon}{4} + \frac{\epsilon}{4} + \frac{3}{4}(\frac{\epsilon}{4} + \frac{\epsilon}{4}) = 0.81\epsilon$ i.e. tighter than 0.83ϵ.

In a more general setting, solving this problem requires us to make groups of the mechanisms with overlapping attributes and make sure that each group is amplified using distinct amplification factors.

Problem 3. *Consider an MCAR mechanism M_Φ and a sequence of j mechanisms $M_1, \ldots, M_j$ with DP guarantees of $\epsilon_1, \ldots, \epsilon_j$ to D, where M_i computes a marginal over attributes $\mathcal{A}_i$. We would like to find amplification attribute sets $\{\mathcal{A}_{M_1} \subseteq \mathcal{A}_1, \ldots, \mathcal{A}_{M_j} \subseteq \mathcal{A}_k\}$ and their corresponding amplification factor $p_1, \ldots, p_j$ for $M_1, \ldots M_j$, that gives the smallest DP cost to the ground truth data $\bar{D}$.*

One way to solve the above problem is by creating valid partitions of marginals and assigning each group in the partition an amplification attribute set such that all groups have disjoint attribute sets and all marginals from the same group are amplified using their own amplification attribute set.

Definition 5.9 (Valid partition). Given DP mechanisms $M_1, \ldots, M_j$ for computing marginals over attribute sets $\mathcal{A}_1, \ldots, \mathcal{A}_j$, a partition of these mechanisms $P = \{G_1, \ldots G_i\}$ is considered valid if it satisfies these conditions: (1) All mechanisms in the same partition are amplified with the same set of amplification attribute set and with the same amplification factors; and (2) The amplification attribute sets of all partitions are disjoint.

The privacy cost for a valid partition is $\bar{\epsilon} = \sum_{G_l \in P} p_{\mathcal{A}_{G_l}} \sum_{M_j \in G_l} \epsilon_j$, where $\mathcal{A}_{G_l}$ is the amplification attribute set for the mechanisms grouped into G_l and $p_{\mathcal{A}_{G_l}}$ is the corresponding amplification factor.

A valid partition ensures each group's amplification attribute set is disjoint, ensuring each attribute is considered only once. To solve Problem 3, we select the partition with the least privacy cost that is also valid. Thus, all DP sub-mechanisms are amplified using the best disjoint attribute set. However, enumerating all valid partitions is intractable due to the exponential number of possibilities[1]. If the cardinality of $\mathcal{A}$ and the number of DP mechanisms j are small, then one can enumerate all possible solutions and choose the best one. However, for datasets with a large number of attributes, we show an initial pruning method to trim away bad solutions using Lemma 5.10.

Lemma 5.10. *A valid partition P solution to Problem 3 should have a non-empty amplification attribute set for all group $G \in P$.*

Proof. Suppose there are a sequence of DP sub-mechanisms $M_1, \ldots, M_j$ with privacy cost of $\epsilon_1, \ldots, \epsilon_j$. A valid partition P over $M_1, \ldots, M_j$ consists of a group G that has an empty amplification attribute set. Then the amplification factor of group G will be $p_G = 1$ and the overall privacy cost of this partition can be calculated using Definition 5.9.

$$\begin{aligned}\bar{\epsilon}_P &= \sum_{G_l \in P} p_{\mathcal{A}_{G_l}} \sum_{M_j \in G_l} \epsilon_j \\ &= \sum_{G_l \in \{P-G\}} p_{\mathcal{A}_{G_l}} \sum_{\mathcal{A}_j \in G_l} \epsilon_j + \sum_{M_j \in G} \epsilon_j \\ &= L + X\end{aligned}$$

, where $L = \sum_{G_l \in \{P-G\}} p_{\mathcal{A}_{G_l}} \sum_{\mathcal{A}_j \in G_l} \epsilon_j$ and $X = \sum_{M_j \in G} \epsilon_j$. We can always define another valid partition P' that has the groups as P but group G has amplification attribute set A. Therefore, the privacy cost of partition P' can be calculated as:

$$\begin{aligned}\bar{\epsilon}_{P'} &= \sum_{G_l \in P'} p_{\mathcal{A}_{G_l}} \sum_{M_j \in G_l} \epsilon_j \\ &= \sum_{G_l \in \{P'-G\}} p_{\mathcal{A}_{G_l}} \sum_{\mathcal{A}_j \in G_l} \epsilon_j + p_A \sum_{M_j \in G} \epsilon_j \\ &= L' + p_A X\end{aligned}$$

, where $L' = \sum_{G_l \in \{P'-G\}} p_{\mathcal{A}_{G_l}} \sum_{\mathcal{A}_j \in G_l} \epsilon_j$. Note that as amplification attribute set A is used for G in P', the privacy cost of the other groups can at most be increased by p_A. Assuming the worst case, $L' = p_A L$ and that the privacy cost of P is always higher than that of P'.

$$\begin{aligned}L + X - L' - p_A X &\geq 0 \\ L + X - p_A L - p_A X &\geq 0 \\ L(1 - p_A) + X(1 - p_A) &\geq 0 \\ (L + X)(1 - p_A) &\geq 0\end{aligned}$$

The above inequality if always true as the first term is always positive (L and X are privacy costs) and the $p_A \leq 1$ as all missing percentages are ≤ 1. □

For large datasets where we are left with multiple partitions even after pruning, we use a brute force search as described in Algorithm 4. In Line 1, we enumerate all possible disjoint amplification attribute sets that we can make from $\mathcal{A}$ and store in a variable $\mathcal{P}_{\mathcal{A}}$. Then, we loop through each possible disjoint attribute set $P_{\mathcal{A}} \in \mathcal{P}_{\mathcal{A}}$, and calculate the cost of each amplification attribute set $\mathcal{A}_l \in P_{\mathcal{A}}$ in Line 2-3 using Proposition 5.4. We initialize the cost of the partition $c_{P_{\mathcal{A}}} = 0$ in Line 4. Then we loop through each pair of marginal S_i and its corresponding DP mechanism (M_i, ϵ_i) and find the candidate amplification attribute sets that are contained in the marginal S_i in Line 6. If no such candidate set is valid for S_i, then we can prune the entire partition $P_{\mathcal{A}}$ using Lemma 5.10 and loop back to Line 2. Otherwise, in Line 7-8, we find the candidate attribute set which has the best amplification cost for M_i and add its corresponding cost the final cost $c_{P_{\mathcal{A}}}$. Finally, the partition with the minimum sum cost is returned in Line 10. In Example 5.11 we show how Algorithm 4 can be used to find the valid partition for our running example with the lowest privacy cost.

Example 5.11. Consider the same setup in Example 5.8. There are total of 4 attributes and Algorithm 4 starts by enumerating all 15 possible disjoint amplification attribute sets — {State | Occupation | Gender | Income}, {State Occupation | Gender | Income}, ... , {State Occupation Gender | Income}, ... , {State Occupation Gender Income}. We then iterate through each of these disjoint sets. Let's consider the disjoint sets $\mathcal{A}_1 : \{State\}, \mathcal{A}_2 : \{Occupation\}, \mathcal{A}_3 : \{Gender\}, \mathcal{A}_4 : \{Income\}$. For each attribute set we calculate its corresponding amplification factor, $c_{\mathcal{A}_1} : (1-\phi_{State}) = 3/4, c_{\mathcal{A}_2} : (1-\phi_{Occupation}) = 0, c_{\mathcal{A}_3} : (1-\phi_{Gender}) = 3/4, c_{\mathcal{A}_4} : (1-\phi_{Income}) = 3/4$. We then iterate through all marginals and choose the best amplification attribute for each marginal. Therefore, M_1 is amplified using $\mathcal{A}_1$, M_2 using $\mathcal{A}_2$, M_3 and M_4 both using $\mathcal{A}_3$. The final privacy cost therefore is $\bar{\epsilon} = \frac{3}{4}\frac{\epsilon}{4} + \frac{\epsilon}{4} + \frac{3}{4}\frac{\epsilon}{4} + \frac{3}{4}\frac{\epsilon}{4} = 0.81\epsilon$. This partition also happens to be the best partition among the 15 partitions.

[1]The total number of partitions of a set is given by the Bell number.

Algorithm 4 Optimal amplified privacy cost

Require: Marginals $\mathcal{S} = \{S_1, \ldots, S_j\}$ over attributes $\mathcal{A}$, DP mechanisms $\mathcal{M} = \{(M_1, \epsilon_1), \ldots, (M_j, \epsilon_j)\}$ for $\mathcal{S}$, Missing probabilities $\Phi = \{\Phi_A | A \in \mathcal{A}\}$ for MCAR

1: Find all possible partitions of $\mathcal{A}$ and store in $\mathcal{P}_{\mathcal{A}}$
2: **for** each attribute partition $P_{\mathcal{A}}$ in $\mathcal{P}_{\mathcal{A}}$ **do**
3: Calculate amplification factor $c_{\mathcal{A}_l} = \prod_{A \in \mathcal{A}_l}(1 - \phi_A)$ for $\mathcal{A}_l \in P_{\mathcal{A}}$
4: Initialize disjoint set cost $c_{P_{\mathcal{A}}} = 0$
5: **for** each $S_i \in \mathcal{S}$ with its corresponding $((M_i, \epsilon_i))$ **do**
6: Skip $P_{\mathcal{A}}$ if $\{\mathcal{A}_l \subseteq S_i | \mathcal{A}_l \in P_{\mathcal{A}}\} = \emptyset$
7: Find the best amplification attribute set for M_i: $\mathcal{A}_{l*} \leftarrow argmin_{\mathcal{A}_l \in P_{\mathcal{A}} \wedge \mathcal{A}_l \subseteq S_i} c_{\mathcal{A}_l}$
8: Add the best amplified cost $c_{P_{\mathcal{A}}} = c_{P_{\mathcal{A}}} + \epsilon_i \cdot c_{\mathcal{A}_{l*}}$
9: **end for**
10: **end for**
11: Return the attribute partition $P_{\mathcal{A}}$ with minimum cost $c_{P_{\mathcal{A}}}$

Use case 3: Privacy amplification for column-wise imputation algorithms. Column-wise imputation algorithms (e.g., KaminoI) learn attributes sequentially in a predefined sequence S over $A_1, \ldots, A_k$. The first attribute S_1 is learned using its observed distribution, while the rest $S_2, \ldots, S_k$ are learned using intermediate models $M_2, \ldots, M_k$ with privacy costs $\epsilon_2, \ldots, \epsilon_k$. At each i^{th} iteration, attribute S_i is learned using intermediate model M_i, taking previously learned attributes $S_{:i}$ as feature input. After training, M_i is used to sample the synthetic dataset and impute incomplete values of $D[S_i]$. Thus, for the next model M_{i+1}, previously learned attributes are either complete or imputed. The total number of complete rows fed to an intermediate model M_i depends solely on the complete values in $D[S_i]$, leading to amplified privacy for ground truth data $\bar{\epsilon_i} = p_{S_i}\epsilon_i$, where $p_{S_i} = 1 - \phi_i$. However, we note that for training model M_i, every attribute $S_{:i-1}$ is considered for amplification using the same probability as S_i. Since each attribute can be amplified only once, privacy for ground truth is calculated accordingly.

Theorem 5.12. *Consider an MCAR mechanism M_Φ and a sequence of attributes S and k mechanisms $M_1, \ldots, M_k$ with privacy cost of $\epsilon_1, \ldots, \epsilon_k$ to D, where M_i is an intermediate model that trains A_i as target and $A_{:i}$ as features. If model M_i is used for imputation of attribute A_i, then the overall process offers DP to the ground truth data $\bar{D}$ at a cost of $\bar{\epsilon} = p_{S_j}\epsilon_j + \sum_{i=1, i\neq j}^{k} \epsilon_i$, where $j = \max_{i=1}^{k}(1 - \phi_i)$.*

Proof. As all cells of an attribute are imputed at previous iterations, the number of complete rows fed to a model at the ith iteration depends on the missing cells of the S_i attribute. The amplification for the ith model can therefore be calculated at $\epsilon_i p_{S_i}$, where $p_{S_i} = 1 - \phi_i$ is the amplification factor of attribute S_i. To calculate the overall privacy of the algorithm, we would like to compose the amplified privacy of all sub models. However, the model M_i trains on every attribute $S_{:i-1}$ and each attribute uses the same probability as S_i. Furthermore as each attribute can only be used once, we can only amplify a single model. Hence, we choose to amplify the model with the maximum amplification M_j where $j = \max_{i=1}^{k}(1 - \phi_i)$ and the other models are left un-amplified. □

Table 1: Dataset Characteristics

Dataset	Cardinality	#Numerical Attr	#Categorical Attr
Adult	32561	5	10
Bank	45211	3	14
BR2000	38000	3	11
National	15012	6	14

Example 5.13. Consider the incomplete dataset from Figure 3. Lets assume we have a column based imputation algorithm learns this dataset. Each attribute is learnt using the equal privacy budget $\frac{\epsilon}{4}$. The attributes 'State', 'Gender' and 'Income' has an amplification factor $\frac{3}{4}$ whereas the attribute 'State' has a factor of 1. Therefore, the amplified privacy budget is $\bar{\epsilon} = \frac{3}{4}\frac{\epsilon}{4} + 3\frac{\epsilon}{4} = 0.9375\epsilon$.

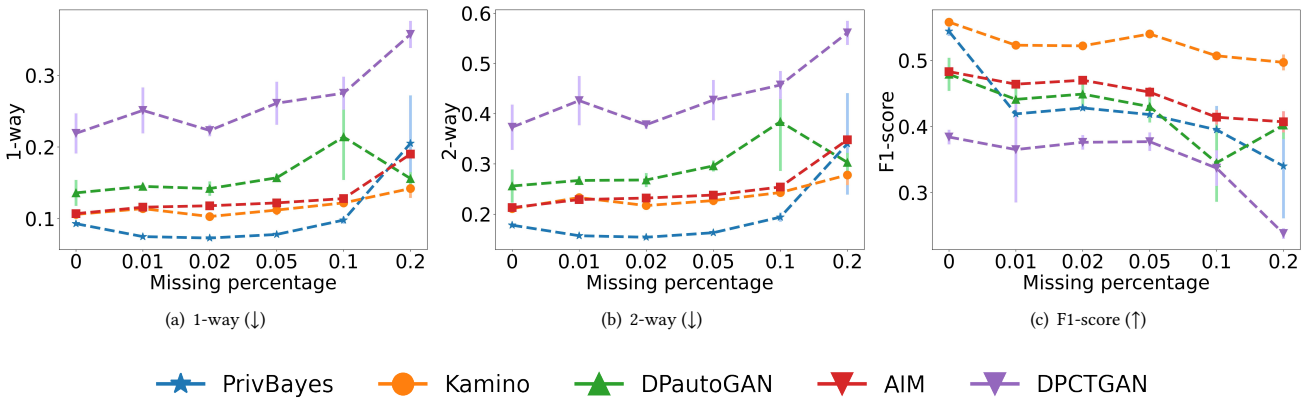
Discussion on MAR and MNAR. MAR and MNAR mechanisms model probabilities of missing data conditioned upon non-missing values from other columns (MAR) or missing values from the same column in the dataset (MNAR). Unfortunately, verifying the presence of MAR and MNAR missing types in the dataset lacks a foolproof method, as it may rely on observed and unobserved variables and their interactions (as discussed in Section 2). Due to the conditional nature of these missing types, achieving privacy amplification with MAR or MNAR mechanisms is challenging. Each row's probability of having missing values may vary due to conditional dependence on dataset values. While maximum probabilities or noisy upper bounds can be used for privacy amplification if these probabilities are known or privately learned, they are often associated with multiple patterns and complex relationships. For instance, in the simple example depicted in Figure 3, rows with the highest income value (80k) are missing. Here, the probability of a row having a missing value depends directly on the probability of a row having an income of 80k. To calculate missing probabilities associated with MNAR, not only the highest value in the column but also its probability of occurrence needs to be determined. Without distribution assumptions, such calculations cannot be achieved with DP [16]. Amplification extensions for MAR and MNAR remain a topic for future research.

6 EVALUATION

We thoroughly experiment with DP synthetic data generation algorithms on missing data. First, we demonstrate how existing DP methods are affected by varying amount of missing data. Next, we evaluate the effectiveness of our proposed adaptive recourse methods and analyze the impact of varying missing data percentages, missing mechanisms, and privacy budgets for each method. Finally, we show how missingness amplifies privacy for ground truth data.

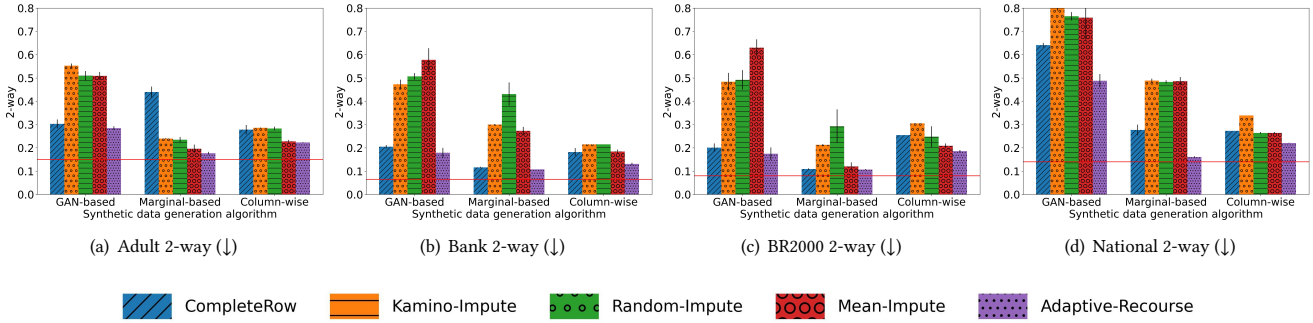
6.1 Experimental Setup

Datasets. We run our experiments on four tabular datasets as described in Table 1: (i) Adult dataset [25], which contains information about 32561 individuals from the 1994 US Census (ii) Bank dataset [64], which has 45211 rows about direct marketing campaigns of a Portuguese banking institution (iii) BR2000 [97], which consists of 38,000 census records collected from Brazil in the year 2000, (iv) National dataset [84] from NIST Diverse Community



(a) 1-way (↓) (b) 2-way (↓) (c) F1-score (↑)

Figure 4: The effect of missing data on DP synthetic data generation algorithms.



(a) Adult 2-way (↓) (b) Bank 2-way (↓) (c) BR2000 2-way (↓) (d) National 2-way (↓)

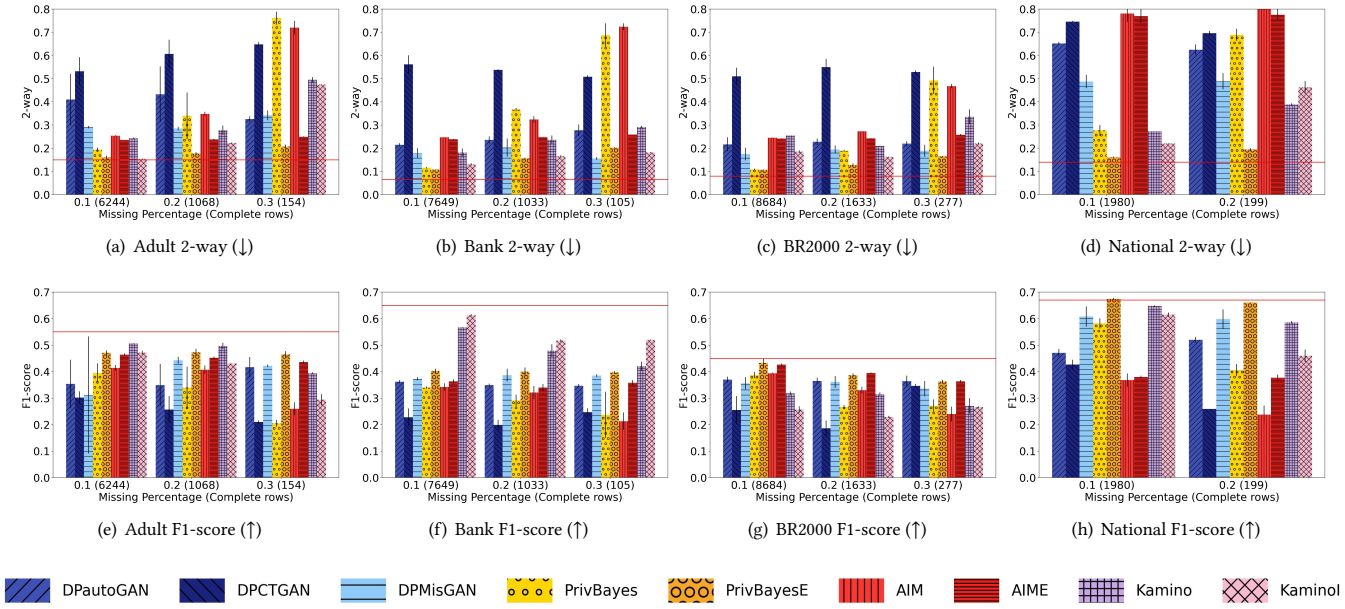
Figure 5: Comparing all strategies to deal with missing data. Adaptive recourse results in the best performance, followed by the complete row approach. The red line represents the best no-missing baseline.

Excerpts which contains information about 15012 individuals US census data. Each dataset has a combination of numerical and categorical columns which are pre-processed according to the synthetic data generation algorithm as discussed in their respective research paper – numerical attributes are discretized into 10 uniform bins or scaled between 0 to 1 and the categorical attributes are encoded using one-hot or ordinal encoding.

Baselines. We consider several existing differentially private data generation methods that do not consider missing data: PrivBayes [97] and AIM [58] from statistical approaches, DPCTGAN [32] and DPautoGAN [82] from deep learning techniques, and Kamino [35], which is a mixed approach. These methods have been published in well-known conferences with their code readily available. For each method, we construct the following baselines to deal with missing data as discussed in Section 4.1. The first baseline referred by its original name combines the original method with the complete row-only approach. Next, we construct baselines for the imputation first approach, which first imputes the data and then generates the synthetic data. We initially considered differentially private data imputation using k-means [23] and OLS regression [24], but they fail in working with categorical columns and require a large privacy budget for the imputation process itself. Hence, we adopt the following imputation methods: (i) random imputation, which fills missing values randomly from the attribute domain; (ii) mean imputation, replacing missing numerical attribute values with the mean and sampling from the probability distribution for categorical attributes; and (iii) Kamino imputation, which leverages intermediate models from Kamino for imputation and shares similar model as Holoclean [2, 90]. Random impute requires no privacy budget, but the other two require splitting of the privacy budget for imputation and for the data generation algorithm.

Parameters. The utility of the algorithms replies upon multiple hyperparameters, and we try our best to tune these parameters by running grid searches. Although we don't consider privacy costs for tuning, it's crucial to tune these parameters privately in practice [63, 71]. Deep learning methods typically require a larger privacy budget for meaningful results. Hence, we assign PrivBayes and Kamino $\epsilon = 1$, and GAN approaches $\epsilon = 3$ by default. Adaptive methods maintain the same budget as their non-adaptive counterparts. PrivBayes operates under pure DP ($\delta = 0$), while others employ approximate DP. The δ value is approximately set one magnitude lower than $1/|D|$ to the nearest exponent of 10.

Missing data. We implement a pipeline that can generate different categories of missing data at different missing percentages, using the approach from Muzellec et al. [65]. MCAR missing data is generated by masking the original dataset using a realization



(a) Adult 2-way (↓) (b) Bank 2-way (↓) (c) BR2000 2-way (↓) (d) National 2-way (↓)

(e) Adult F1-score (↑) (f) Bank F1-score (↑) (g) BR2000 F1-score (↑) (h) National F1-score (↑)

Figure 6: Adaptive methods (DPMisGAN, PrivBayesE, KaminoI) vs their respective baselines using the complete row approach (DPautoGAN, PrivBayes, Kamino) with MCAR missingness at $\epsilon = 1$. Algorithms of the same category are colored with the same shade. The adaptive methods result in better quality synthetic data. The red line denotes the best no missing baseline.

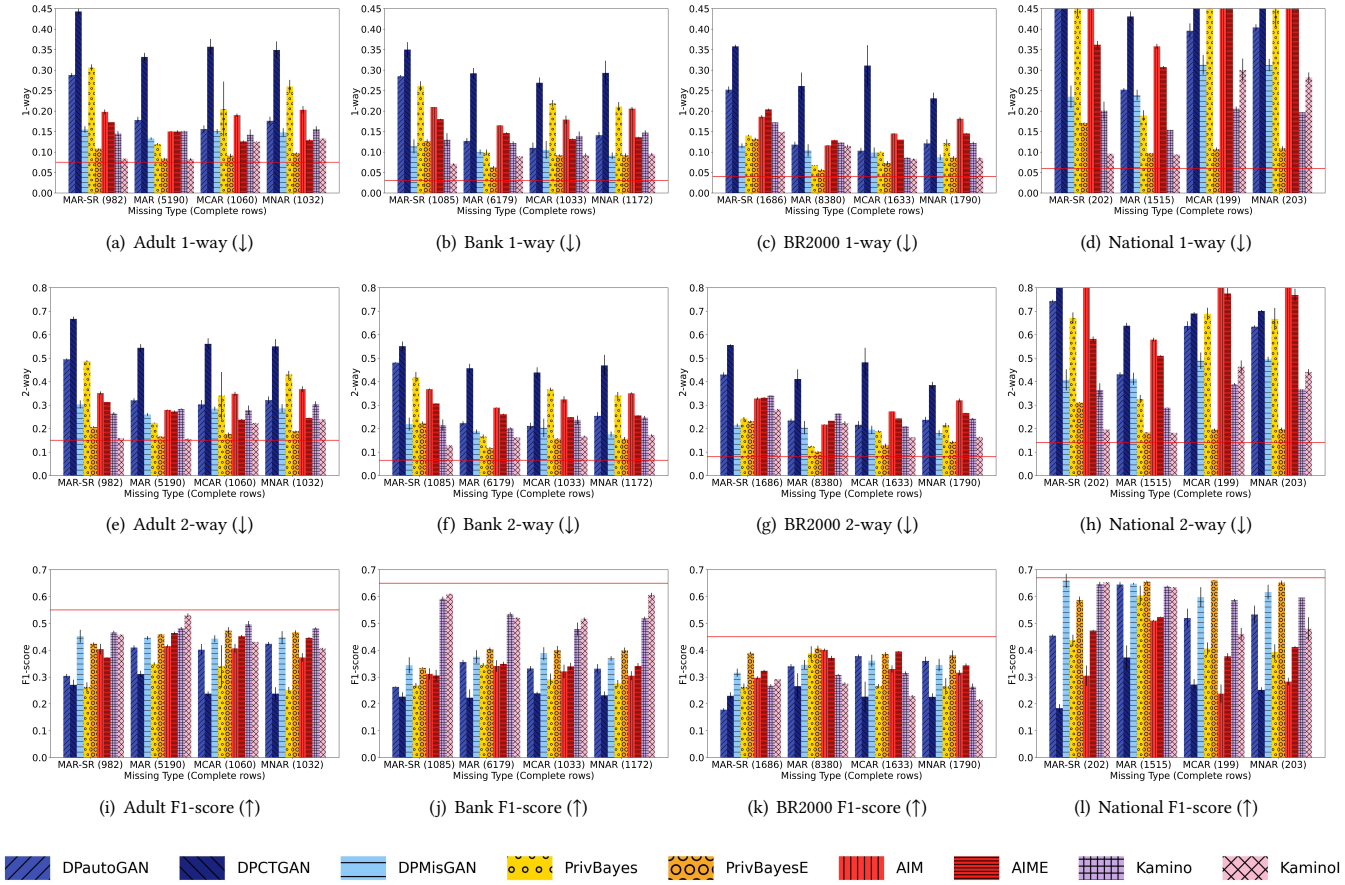
of a Bernoulli variable with a fixed parameter such that there is exactly the required number of missing values. To generate MAR data, we first use 50% of the attributes in the dataset as features for a logistic regression model. The other attributes then have missing values according to random weights in the logistic model. A bias term is fitted using line search to attain the desired proportion of missing values. The MNAR approach works similarly to MAR, with the difference that the 50% non-missing attributes are masked by an MCAR mechanism. This imposes the logistic model to depend potentially on missing values, hence enforcing MNAR missingness. As 50% of the columns in the MAR mechanism has complete values, it comparatively has more complete rows in comparison to MCAR and MNAR. We run experiments on all kinds of missing types for every dataset and go up to 30% missing values except the national dataset, where we stop at 20% due to the lower number of rows in the original dataset.

Metrics. We use two utility metrics for evaluation. The first metric is k-way marginal distance. We calculate all k-sized combinations of attributes in the dataset [35, 97], and then report the average variational distance between the true vs the synthetic marginals. For each k-sized combination set of attributes in $A \in \mathcal{A}$, we calculate the marginal, $h : \mathcal{D} \rightarrow \mathbb{R}^{|\mathcal{D}(A)|}$ and report the average variational distance between as $\max_{a \in \mathcal{D}(A)} |h(D')[a] - h(\bar{D})[a]|$ where D' and $\bar{D}$ are the synthetic data and ground true data respectively [35, 97]. In our evaluation, we set $k = 1, 2$. Smaller values of this metric show more closeness between the true and synthetic data. Our second metric is model training. We consider 9 classification models (LogisticRegression, AdaBoost, GradientBoost, XGBoost, RandomForest, BernoulliNB, DecisionTree, Bagging, and MLP) to classify each attribute in the dataset (e.g., income is more than 50k or not, the loan should be given or not to user) using all other attributes as features. Each target attribute is processed to be a binary attribute, and we try to balance the two classes as much as possible. The quality of the learning task is represented by the average of all models across all attributes. The F1 score is reported for learning quality. Each model is trained using 70% of the synthetic data, and the F1 score is evaluated using 30% of the true data [35]. Larger values of this metric show better performance. For all our experiments, we run 3 times and report the mean with standard deviation.

6.2 Experimental Findings

We evaluate the impact of missing values on DP synthetic data generation algorithms with different missing mechanisms and varying missing data from 1% to 30%. The quality of synthetic data generated by all algorithms degrades drastically, particularly when having >10% of missing data.

6.2.1 Benchmark Existing DP Methods on Missing Data. We demonstrate the impact of missing values on DP synthetic data generation algorithms. Figure 4 shows the performance of the baselines on the Adult dataset with varying levels of missing completely at random (MCAR) data (x-axis). Our results indicate that missing data up to 5% minimally affects the quality of the synthetic data with 1-way and 2-way metrics experiencing 5-19% impact, and F1-score experiencing 1-13% impact. However, as missing data increases, the quality of synthetic data generated by all algorithms degrades drastically. With 20% missing data, the 1-way metric is affected by 14-190%, the 2-way by 18-147%, and the F1-score is impacted by 6-28%. With high amounts of missing data ($\geq$ 10%), PrivBayes and DPCTGAN exhibit the most decrease in utility. However, Kamino



(a) Adult 1-way (↓) (b) Bank 1-way (↓) (c) BR2000 1-way (↓) (d) National 1-way (↓)

(e) Adult 2-way (↓) (f) Bank 2-way (↓) (g) BR2000 2-way (↓) (h) National 2-way (↓)

(i) Adult F1-score (↑) (j) Bank F1-score (↑) (k) BR2000 F1-score (↑) (l) National F1-score (↑)

Figure 7: Adaptive methods vs non-adaptive methods on different missing mechanisms. The red line denotes the no missing baseline.

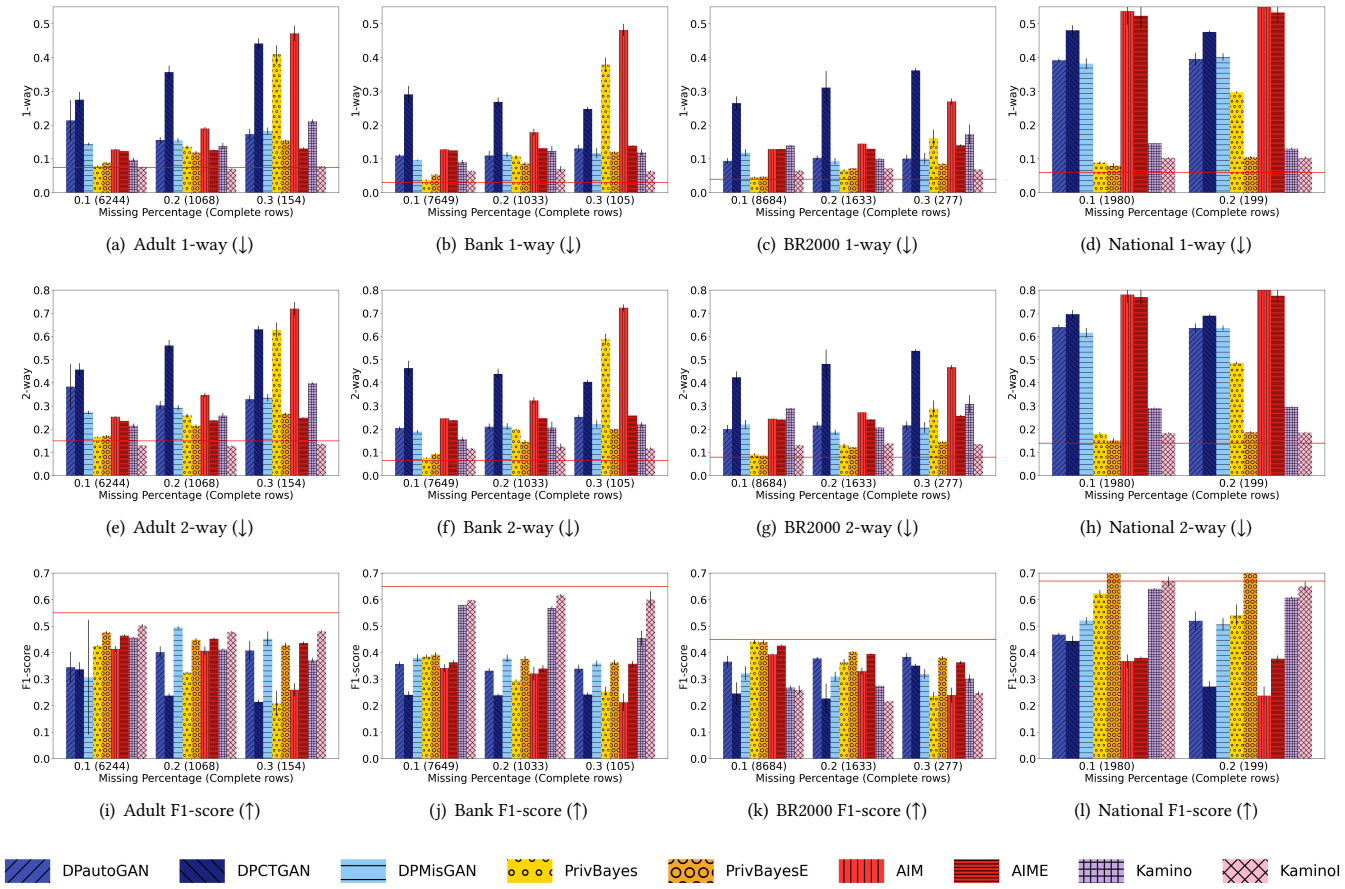
and DPAutoGAN show a consistent level of performance even with more missing data indicating a high degree of stability in the output. The stability of these methods can be attributed to the fact that these methods tend to extract more information from the available data. Kamino learns the functional dependencies of the dataset and tries to preserve them while generating the synthetic data and DPAutoGAN trains an autoencoder as a preprocessing step to learn the low-dimensional statistics of the data.

6.2.2 Evaluate different strategies to deal with missing data. For each existing DP algorithm for generating synthetic data, we compare its respective baseline methods and its adaptive recourse approach. Figure 5 shows the performance of these methods across all four datasets, employing 20% MCAR missing values. The red line denotes the baseline approach's performance without missing values. For mean and Kamino imputation, we allocate three splits of the privacy budget (25%, 50%, and 75%) for imputation and plot the average and standard deviation across all splits, reserving the rest for generating the synthetic dataset. We observe that no single imputation strategy consistently outperforms others, with the adaptive recourse approach generally yielding the best results. Hence, allowing learning and imputation to happen together is crucial. Additionally, the complete row approach often performs second best, serving as our baseline for subsequent experiments.

6.2.3 Evaluate Adaptive Recourse Approach. We evaluate adaptive recourse approaches at various experimental configurations.

Varying missing percentage. In this experiment, we compare the best baseline from the previous experiment that uses the complete row approach with its corresponding adaptive recourse approach from Section 4.2. We repeat our experiment on four different datasets with varying amount of MCAR data at two privacy levels ($\epsilon = 1, 3$).

In Figure 6, we plot the algorithms in different shades of color depending on their type. We observe in general that adaptive recourse approaches (DPMisGAN, PrivBayesE, AIME, and KaminoI) result in significantly better quality synthetic data compared to their baseline with the complete row only approach (DPautoGAN, DPCTGAN, PrivBayes, AIM and Kamino). Across all datasets, the 1-way scores are improved by up to 68%, 2-way by up to 66% and F1-scores of up to 24%. Furthermore, we observe that the adaptive



(a) Adult 1-way (↓) (b) Bank 1-way (↓) (c) BR2000 1-way (↓) (d) National 1-way (↓)

(e) Adult 2-way (↓) (f) Bank 2-way (↓) (g) BR2000 2-way (↓) (h) National 2-way (↓)

(i) Adult F1-score (↑) (j) Bank F1-score (↑) (k) BR2000 F1-score (↑) (l) National F1-score (↑)

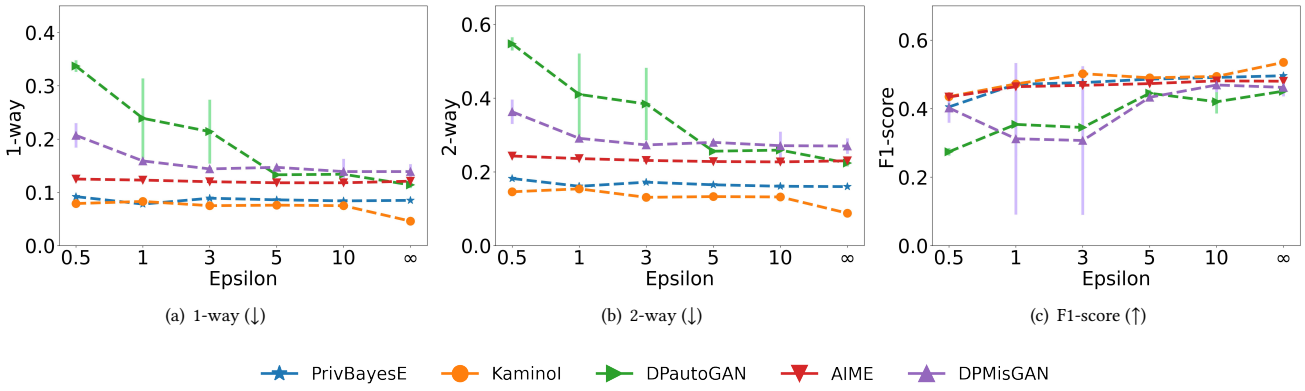
Figure 8: Baselines vs Adaptive methods at $\epsilon = 3$. The adaptive methods perform better than their respective baselines.

methods often achieve the same utility as the no missing baseline with 10% missing data (e.g., top left subfigure Adult 2-way for KaminoI and PrivBayesE at 10% missing data).

Varying missing mechanisms. In Figure 7, we repeat our experiment for all 4 datasets at 10% missing data for all missing mechanisms. The adaptive recouse approaches(DPMisGAN, PrivBayesE, AIME and KaminoI) beat their non-adaptive complete-row baselines across all missing mechanisms. As missing values are added only to half of the attributes, missing at random (MAR) has more complete rows (the number in brackets on the x-axis labels) as compared to the other mechanisms. It is interesting to note that if we increase the missing percentage for MAR and plot it with same rows (MAR-SR), the algorithms start performing poorly. Hence, we make the conclusion that the number of complete of rows makes a more vital impact compared to the missing mechanism itself.

Varying privacy budget. We show the impact of the privacy budget in Figure 9 for adaptive methods with 10% MCAR missing data on Adult. We vary the privacy budget $\epsilon \in [0.5, 1, 3, 5, 10, \infty]$ where $\epsilon = \infty$ refers to the non-private run. First, we note that increasing the privacy budget improves the utility of the synthetic dataset across all algorithms. Second, we observe that PrivBayesE and KaminoI outperformed GAN-based approaches at all privacy budgets. Similar observations are found in Figure 6 and Figure 7. The poor performance of the GAN-based approaches is primarily due to the requirement for specific preprocessing steps and extensive hyperparameter tuning, which was challenging. Third, we find that there is no clear winner for all datasets, all utility metrics, and all privacy budgets, which is consistent with the observations from prior benchmarks on DP synthetic data generation [83]. However, we observe that PrivBayesE performs better at smaller epsilon values ($\epsilon \leq 1$), but KaminoI has better performance with higher epsilon ($\epsilon > 1$). We attribute this phenomenon to the fact that KaminoI trains multiple intermediate models, and these models generally require a higher privacy budget. Therefore, we recommend PrivBayesE at a low privacy regime ($\epsilon \leq 1$) and KaminoI at a higher privacy regime ($\epsilon > 1$).

In addition, we observe that PrivBayesE outperforms others at the 2-way tasks (Figure 6 top row), and KaminoI demonstrates superior performance in the F1-score metric even at a low privacy regime ($\epsilon = 1$, Figure 6 bottom row, except BR2000). This shows that depending on the data sets and the downstream tasks different methods perform best. Hence, we also recommend to an end user



(a) 1-way (↓) (b) 2-way (↓) (c) F1-score (↑)

Figure 9: Varying privacy budget and comparing utilities of our approaches

Table 2: Amplified privacy for ground truth data.

Dataset	MCAR missing %				
	0.1	0.2	0.3	0.4	0.5
Adult	0.88	0.77	0.65	0.47	0.44
BR2000	0.83	0.68	0.55	0.41	0.31

to test all methods for their specific data sets and downstream tasks and assess result quality in the way we have done. Then they can select the best method for their case, but this ignores privacy cost of algorithm selection. Generally, we recommend marginal-based approach like PrivBayesE for k-way tasks and KaminoI for ML tasks.

6.2.4 Amplification Due To Missingness. In this experiment, we show the amplified privacy budget for ground truth data. We experiment with PrivBayesE that runs on the incomplete dataset with 10 - 50% MCAR missing data. For each run, we allocate a privacy budget of $\epsilon = 1$ and observe the marginals calculated by PrivBayesE. We assume a uniform privacy budget for each marginal calculated by PrivbayesE and run Algorithm 4 to calculate the best valid partition of these marginals. In Table 2, we plot the amplified privacy cost $\bar{\epsilon}$ based on the best partition found by the algorithm. We repeat the experiment for two datasets – Adult and BR2000. Our results show that the amplified privacy cost decreases almost linearly from 0.88x to 0.44x for Adult and 0.83x to 0.31x for the BR2000 dataset.

7 RELATED WORK

Differentially private synthetic data generation has been studied vastly in prior literature [15, 31, 69, 98]. Prior works generate synthetic data via statistical approaches which estimate low-dimensional marginal distributions [76, 91], deep learning approaches [33, 45], or the combination of the two [35]. However, all of these algorithms focus on the no missing data setting. Some prior work look into missing data imputation for private datasets but are either not in the differential privacy setting [40, 42, 72] or do not support generating synthetic data as a part of their work [23, 24, 49]. Patki et al. [72] propose the synthetic data vault framework, which identifies and repairs inconsistencies in the generated synthetic data from incomplete data but does not consider any privacy guarantees. Their approach uses low-way marginals and learns them using Gaussian copulas that may be enhanced using our partial marginal observation approach if made private. Huang et al. [40] and Jagannathan et al. [42] imputation of missing data as a cleaning algorithm for private datasets but consider privacy definitions of k-anonymity and cryptographic distributive computing, respectively. Other works focus on privately imputing missing values. The existing differentially private solutions [23, 24, 49] focus on data imputation and are difficult to adapt for the synthetic data generation problem. PrivateClean [49] requires a human in the loop that can specify the user-defined specific imputation functions to clean the database. The k-means based word by Clifton et al. [23] and imputation first approach in Das et al. [24] use part of the privacy budget to impute the missing values using OLS regression. We note that such OLS regression and k-means models cannot be applied to categorical attributes, and using them for imputation incurs a significant amount of privacy budget and leaves little budget for data synthesis.

8 CONCLUSION

In conclusion, our research paper presents a comprehensive study on differentially private synthetic data generation algorithms for private datasets with missing values. Our proposed adaptive recourse methods outperform classical approaches and strike a balance between privacy and utility. We also provide techniques for calculating privacy bounds and demonstrate the effectiveness of our methods through extensive experiments on real-world datasets. Our findings have important implications for privacy-preserving data sharing and analysis, and can facilitate the development of more effective methods for generating synthetic data in various practical applications.

REFERENCES

[1] 2016-04-27. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). *OJ* (2016-04-27).
[2] Online. HoloClean code. https://github.com/HoloClean/holoclean/

[3] Martín Abadi, Andy Chu, Ian J. Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep Learning with Differential Privacy. In *CCS*. ACM, 308–318.

[4] Nazmiye Ceren Abay, Yan Zhou, Murat Kantarcioglu, Bhavani Thuraisingham, and Latanya Sweeney. 2018. Privacy preserving synthetic data release using deep learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 510–526.

[5] John M. Abowd. 2018. The U.S. Census Bureau Adopts Differential Privacy. In *KDD*. 2867.

[6] Gergely Acs, Luca Melis, Claude Castelluccia, and Emiliano De Cristofaro. 2018. Differentially private mixture of generative neural networks. *IEEE Transactions on Knowledge and Data Engineering* 31, 6 (2018), 1109–1121.

[7] Anish Agarwal and Rahul Singh. 2021. Causal inference with corrupted data: Measurement error, missing values, discretization, and differential privacy. *arXiv preprint arXiv:2107.02780* (2021).

[8] Elena Andreou, Eric Ghysels, and Andros Kourtellos. 2013. Should macroeconomic forecasters use daily financial data and how? *Journal of Business & Economic Statistics* 31, 2 (2013), 240–251.

[9] Rebecca R Andridge and Roderick JA Little. 2010. A review of hot deck imputation for survey non-response. *International statistical review* 78, 1 (2010), 40–64.

[10] Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein generative adversarial networks. In *International conference on machine learning*. PMLR, 214–223.

[11] Brooke Auxier, Lee Rainie, Monica Anderson, Andrew Perrin, Madhu Kumar, and Erica Turner. 2019. Americans and Privacy - Concerned Confused and Feeling Lack of Control Over Their Personal Information. *Pew Research Center* (2019).

[12] Borja Balle, Gilles Barthe, and Marco Gaboardi. 2018. Privacy amplification by subsampling: Tight analyses via couplings and divergences. *Advances in Neural Information Processing Systems* 31 (2018).

[13] Boaz Barak, Kamalika Chaudhuri, Cynthia Dwork, Satyen Kale, Frank McSherry, and Kunal Talwar. 2007. Privacy, accuracy, and consistency too: a holistic solution to contingency table release. In *PODS*. 273–282.

[14] Raef Bassily, Adam D. Smith, and Abhradeep Thakurta. 2014. Private Empirical Risk Minimization: Efficient Algorithms and Tight Error Bounds. In *FOCS*. 464–473.

[15] Claire McKay Bowen and Fang Liu. 2020. Comparative Study of Differentially Private Data Synthesis Methods. *Statist. Sci.* 35, 2 (May 2020), 280–307. https://doi.org/10.1214/19-sts742

[16] Mark Bun, Kobbi Nissim, Uri Stemmer, and Salil Vadhan. 2015. Differentially private release and learning of threshold functions. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*. IEEE, 634–649.

[17] U.S. Census Bureau. Accessed on 2020-11-30. LEHD Origin-Destination Employment Statistics (2002-2017). https://onthemap.ces.census.gov/

[18] Thee Chanyaswad, Changchang Liu, and Prateek Mittal. 2019. Ron-gauss: Enhancing utility in non-interactive private data release. *Proceedings on Privacy Enhancing Technologies* 2019, 1 (2019), 26–46.

[19] R. Chawla. 2019. Deepfakes : How a pervert shook the world. *International Journal for Advance Research and Development* 4 (2019), 4–8.

[20] Dingfan Chen, Tribhuvanesh Orekondy, and Mario Fritz. 2020. Gs-wgan: A gradient-sanitized approach for learning differentially private generators. *Advances in Neural Information Processing Systems* 33 (2020), 12673–12684.

[21] Qingrong Chen, Chong Xiang, Minhui Xue, Bo Li, Nikita Borisov, Dali Kaafar, and Haojin Zhu. 2018. Differentially Private Data Generative Models. *CoRR* abs/1812.02274 (2018).

[22] Rui Chen, Qian Xiao, Yu Zhang, and Jianliang Xu. 2015. Differentially Private High-Dimensional Data Publication via Sampling-Based Inference. In *SIGKDD*. 129–138.

[23] Chris Clifton, Eric J. Hanson, Keith Merrill, and Shawn Merrill. 2022. Differentially Private k-Nearest Neighbor Missing Data Imputation. *ACM Trans. Priv. Secur.* 25, 3 (2022), 16:1–16:23. https://doi.org/10.1145/3507952

[24] Soumojit Das, Jorg Dreschler, Keith Merrill, and Shawn Merrill. 2022. Imputation under Differential Privacy. *CoRR* abs/2206.15063 (2022). https://doi.org/10.48550/arXiv.2206.15063

[25] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml

[26] Cynthia Dwork. 2006. Differential Privacy. In *ICALP*, Vol. 4052. Springer, 1–12.

[27] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. 2006. Our Data, Ourselves: Privacy Via Distributed Noise Generation. In *EUROCRYPT*, Vol. 4004. Springer, 486–503.

[28] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating Noise to Sensitivity in Private Data Analysis. In *Proceedings of the 3rd Conference on Theory of Cryptography (TCC '06)*. 265–284.

[29] Cynthia Dwork and Aaron Roth. 2014. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends in Theoretical Computer Science* 9, 3-4 (2014), 211–407.

[30] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. 2014. RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response. In *CCS*. ACM, 1054–1067.

[31] Liyue Fan. 2020. A Survey of Differentially Private Generative Adversarial Networks. In *The AAAI Workshop on Privacy-Preserving Artificial Intelligence*.

[32] Mei Ling Fang, Devendra Singh Dhami, and Kristian Kersting. 2022. Dp-ctgan: Differentially private medical data generation using ctgans. In *International Conference on Artificial Intelligence in Medicine*. Springer, 178–188.

[33] Lorenzo Frigerio, Anderson Santana de Oliveira, Laurent Gomez, and Patrick Duverger. 2019. Differentially Private Generative Adversarial Networks for Time Series, Continuous, and Discrete Open Data. In *SEC*. 151–164.

[34] Chang Ge, Xi He, Ihab F. Ilyas, and Ashwin Machanavajjhala. 2019. APEx: Accuracy-Aware Differentially Private Data Exploration. In *SIGMOD*. 177–194.

[35] Chang Ge, Shubhankar Mohapatra, Xi He, and Ihab F Ilyas. 2021. Kamino: Constraint-aware differentially private data synthesis. *Proceedings of the VLDB Endowment* 14, 10 (2021), 1886–1899.

[36] Kylie Getz, Rebecca A Hubbard, and Kristin A Linn. 2023. Performance of Multiple Imputation Using Modern Machine Learning Methods in Electronic Health Records Data. *Epidemiology* (2023), 10–1097.

[37] Andy Greenberg. 2016. Apple's 'Differential Privacy' Is About Collecting Your Data—But Not *Your* Data. *Wired* (2016).

[38] Rahul Gupta. 2019. Data Augmentation for Low Resource Sentiment Analysis Using Generative Adversarial Networks. In *ICASSP*. IEEE, 7380–7384.

[39] Michael B. Hawes. 2020. Implementing Differential Privacy: Seven Lessons From the 2020 United States Census. *Harvard Data Science Review* (30 4 2020). https://doi.org/10.1162/99608f92.353c6f99 https://hdsr.mitpress.mit.edu/pub/dgg03vo6.

[40] Yu Huang, Mostafa Milani, and Fei Chiang. 2018. PACAS: Privacy-aware, data cleaning-as-a-service. In *2018 IEEE International Conference on Big Data (Big Data)*. IEEE, 1023–1030.

[41] IBM. 2020. Cost of a Data Breach Report. (2020).

[42] Geetha Jagannathan and Rebecca N Wright. 2008. Privacy-preserving imputation of missing data. *Data & Knowledge Engineering* 65, 1 (2008), 40–56.

[43] José M Jerez, Ignacio Molina, Pedro J García-Laencina, Emilio Alba, Nuria Ribelles, Miguel Martín, and Leonardo Franco. 2010. Missing data imputation using statistical and machine learning methods in a real breast cancer problem. *Artificial intelligence in medicine* 50, 2 (2010), 105–115.

[44] Noah M. Johnson, Joseph P. Near, and Dawn Song. 2018. Towards Practical Differential Privacy for SQL Queries. *PVLDB* 11, 5 (2018), 526–539.

[45] James Jordon, Jinsung Yoon, and Mihaela van der Schaar. 2019. PATE-GAN: Generating Synthetic Data with Differential Privacy Guarantees. In *ICLR*.

[46] Nancy E Kass, Marvin R Natowicz, Sara Chandros Hull, Ruth R Faden, Laura Plantinga, Lawrence O Gostin, and Julia Slutsman. 2003. The use of medical records in research: what do patients want? *The Journal of Law, Medicine & Ethics* 31, 3 (2003), 429–433.

[47] Daphne Koller and Nir Friedman. 2009. *Probabilistic Graphical Models - Principles and Techniques*. MIT Press.

[48] Ios Kotsogiannis, Yuchao Tao, Xi He, Maryam Fanaeepour, Ashwin Machanavajjhala, Michael Hay, and Gerome Miklau. 2019. PrivateSQL: A Differentially Private SQL Query Engine. *PVLDB* 12, 11 (2019), 1371–1384.

[49] Sanjay Krishnan, Jiannan Wang, Michael J Franklin, Ken Goldberg, and Tim Kraska. 2016. Privateclean: Data cleaning and differential privacy. In *Proceedings of the 2016 International Conference on Management of Data*. 937–951.

[50] Kamakshi Lakshminarayan, Steven A Harp, Robert P Goldman, Tariq Samad, et al. 1996. Imputation of Missing Data Using Machine Learning Techniques.. In *KDD*, Vol. 96.

[51] Chao Li, Gerome Miklau, Michael Hay, Andrew McGregor, and Vibhor Rastogi. 2015. The matrix mechanism: optimizing linear counting queries under differential privacy. *VLDB J.* 24, 6 (2015), 757–781.

[52] Haoran Li, Li Xiong, Lifan Zhang, and Xiaoqian Jiang. 2014. DPSynthesizer: Differentially Private Data Synthesizer for Privacy Preserving Data Sharing. *Proc. VLDB Endow.* 7, 13 (2014), 1677–1680.

[53] Steven Cheng-Xian Li, Bo Jiang, and Benjamin Marlin. 2019. Misgan: Learning from incomplete data with generative adversarial networks. *arXiv preprint arXiv:1902.09599* (2019).

[54] Roderick JA Little. 1988. A test of missing completely at random for multivariate data with missing values. *Journal of the American statistical Association* 83, 404 (1988), 1198–1202.

[55] Roderick JA Little. 1994. A class of pattern-mixture models for normal incomplete data. *Biometrika* 81, 3 (1994), 471–483.

[56] Roderick JA Little and Donald B Rubin. 2019. *Statistical analysis with missing data*. Vol. 793. John Wiley & Sons.

[57] Yonghong Luo, Ying Zhang, Xiangrui Cai, and Xiaojie Yuan. 2019. E2gan: End-to-end generative adversarial network for multivariate time series imputation. In *Proceedings of the 28th international joint conference on artificial intelligence*. AAAI Press, 3094–3100.

[58] Ryan McKenna, Brett Mullins, Daniel Sheldon, and Gerome Miklau. 2022. AIM: An Adaptive and Iterative Mechanism for Differentially Private Synthetic Data. *CoRR* abs/2201.12677 (2022). arXiv:2201.12677 https://arxiv.org/abs/2201.12677

[59] Ryan McKenna, Daniel Sheldon, and Gerome Miklau. 2019. Graphical-model based estimation and inference for differential privacy. In *ICML*, Vol. 97. 4435–4444.

[60] Frank McSherry. 2009. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In *SIGMOD*, Ugur Çetintemel, Stanley B. Zdonik, Donald Kossmann, and Nesime Tatbul (Eds.). ACM, 19–30.

[61] Ilya Mironov. 2017. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*. IEEE, 263–275.

[62] Ilya Mironov, Kunal Talwar, and Li Zhang. 2019. Rényi Differential Privacy of the Sampled Gaussian Mechanism. *CoRR* abs/1908.10530 (2019). arXiv:1908.10530 http://arxiv.org/abs/1908.10530

[63] Shubhankar Mohapatra, Sajin Sasy, Xi He, Gautam Kamath, and Om Thakkar. 2021. The Role of Adaptive Optimizers for Honest Private Hyperparameter Selection. *arXiv preprint arXiv:2111.04906* (2021).

[64] Sérgio Moro, Paulo Cortez, and Paulo Rita. 2014. A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems* 62 (2014), 22–31.

[65] Boris Muzellec, Julie Josse, Claire Boyer, and Marco Cuturi. 2020. Missing data imputation using optimal transport. In *International Conference on Machine Learning*. PMLR, 7130–7140.

[66] Shinichi Nakagawa and Robert P Freckleton. 2008. Missing inaction: the dangers of ignoring missing data. *Trends in ecology & evolution* 23, 11 (2008), 592–596.

[67] Eric Schulte Nordholt. 1998. Imputation: methods, simulation experiments and practical examples. *International Statistical Review* 66, 2 (1998), 157–180.

[68] AH Noruzman, NA Ghani, and NSA Zulkifli. [n.d.]. Gretel. ai: Open-Source Artificial Intelligence Tool To Generate New Synthetic Data. ([n. d.]).

[69] National Institute of Standards and Technology. 2018. Differential Privacy Synthetic Data Challenge. https://www.nist.gov/ctl/pscr/open-innovation-prize-challenges/past-prize-challenges/2018-differential-privacy-synthetic

[70] Nicolas Papernot, Shuang Song, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Úlfar Erlingsson. 2018. Scalable Private Learning with PATE. In *ICLR*.

[71] Nicolas Papernot and Thomas Steinke. [n.d.]. Hyperparameter Tuning with Renyi Differential Privacy. In *International Conference on Learning Representations*.

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

[73] NhatHai Phan, Yue Wang, Xintao Wu, and Dejing Dou. 2016. Differential privacy preservation for deep auto-encoders: an application of human behavior prediction. In *Thirtieth AAAI Conference on Artificial Intelligence*.

[74] Haoyue Ping, Julia Stoyanovich, and Bill Howe. 2017. DataSynthesizer: Privacy-Preserving Synthetic Datasets. In *SSDBM*. ACM, 42:1–42:5.

[75] Eugenia Politou, Efthimios Alepis, and Constantinos Patsakis. 2018. Forgetting personal data and revoking consent under the GDPR: Challenges and proposed solutions. *Journal of cybersecurity* 4, 1 (2018), tyy001.

[76] Wahbeh H. Qardaji, Weining Yang, and Ninghui Li. 2014. PriView: practical differentially private release of marginal contingency tables. In *SIGMOD*. 1435–1446.

[77] Jalan Rivers, Andrea Nelson, and Lauren Williams. [n.d.]. Synthetic Data Generation with SDV. ([n. d.]).

[78] Donald B Rubin. 1976. Inference and missing data. *Biometrika* 63, 3 (1976), 581–592.

[79] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Joshua Susskind, Wenda Wang, and Russell Webb. 2017. Learning from Simulated and Unsupervised Images through Adversarial Training. In *CVPR*. IEEE Computer Society, 2242–2251.

[80] Shuang Song, Kamalika Chaudhuri, and Anand D. Sarwate. 2013. Stochastic gradient descent with differentially private updates. In *GlobalSIP*. 245–248.

[81] Thomas Steinke. 2022. Composition of Differential Privacy & Privacy Amplification by Subsampling. *arXiv preprint arXiv:2210.00597* (2022).

[82] Uthaipon Tao Tantipongpipat, Chris Waites, Digvijay Boob, Amaresh Ankit Siva, and Rachel Cummings. 2021. Differentially private synthetic mixed-type data generation for unsupervised learning. In *2021 12th International Conference on Information, Intelligence, Systems & Applications (IISA)*. IEEE, 1–9.

[83] Yuchao Tao, Ryan McKenna, Michael Hay, Ashwin Machanavajjhala, and Gerome Miklau. 2021. Benchmarking Differentially Private Synthetic Data Generation Algorithms. *arXiv preprint arXiv:2112.09238* (2021).

[84] Christine Task, Karan Bhagat, Streat Damon, and Gary Howarth. 2022. NIST Diverse Community Excerpts Data. https://doi.org/10.18434/MDS2-2895

[85] Reihaneh Torkzadehmahani, Peter Kairouz, and Benedict Paten. 2019. Dp-cgan: Differentially private synthetic data and label generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 0–0.

[86] Reihaneh Torkzadehmahani, Peter Kairouz, and Benedict Paten. 2020. DP-CGAN: Differentially Private Synthetic Data and Label Generation. *CoRR* abs/2001.09700 (2020).

[87] Stef Van Buuren. 2018. *Flexible imputation of missing data*. CRC press.

[88] Christopher Waites. 2019. PyVacy: Towards Practical Differential Privacy for Deep Learning. *https://github.com/ChrisWaites/pyvacy* (2019).

[89] Oliver Williams and Frank McSherry. 2010. Probabilistic Inference and Differential Privacy. In *NIPS*. 2451–2459.

[90] Richard Wu, Aoqian Zhang, Ihab F. Ilyas, and Theodoros Rekatsinas. 2020. Attention-based Learning for Missing Data Imputation in HoloClean. In *MLSys*.

[91] Xiaokui Xiao, Guozhang Wang, and Johannes Gehrke. 2011. Differential Privacy via Wavelet Transforms. *IEEE Trans. Knowl. Data Eng.* 23, 8 (2011), 1200–1214.

[92] Liyang Xie, Kaixiang Lin, Shu Wang, Fei Wang, and Jiayu Zhou. 2018. Differentially Private Generative Adversarial Network. *CoRR* abs/1802.06739 (2018).

[93] Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, and Kalyan Veeramachaneni. 2019. Modeling tabular data using conditional gan. *Advances in Neural Information Processing Systems* 32 (2019).

[94] Lei Xu and Kalyan Veeramachaneni. 2018. Synthesizing tabular data using generative adversarial networks. *arXiv preprint arXiv:1811.11264* (2018).

[95] Yungang Xu, Zhigang Zhang, Lei You, Jiajia Liu, Zhiwei Fan, and Xiaobo Zhou. 2020. scIGANs: single-cell RNA-seq imputation using generative adversarial networks. *Nucleic acids research* 48, 15 (2020), e85–e85.

[96] Jinsung Yoon, James Jordon, and Mihaela Schaar. 2018. Gain: Missing data imputation using generative adversarial nets. In *International conference on machine learning*. PMLR, 5689–5698.

[97] Jun Zhang, Graham Cormode, Cecilia M. Procopiuc, Divesh Srivastava, and Xiaokui Xiao. 2014. PrivBayes: private data release via bayesian networks. In *SIGMOD*. 1423–1434.

[98] Tianqing Zhu, Gang Li, Wanlei Zhou, and Philip S. Yu. 2017. Differentially Private Data Publishing and Analysis: A Survey. *IEEE Trans. Knowl. Data Eng.* 29, 8 (2017), 1619–1638.