Cost-Efficient and Privacy-Preserving Synthesis of Complex Sensitive Data

Sandeep Suresh
University of Technology Sydney
Sydney, Australia
sandeep.suresh@student.uts.edu.au

Guangsheng Zhang
University of Technology Sydney
Sydney, Australia
Guangsheng.Zhang@student.uts.edu.au

Bo Liu
University of Technology Sydney
Sydney, Australia
bo.liu@uts.edu.au

Barry Drake
University of Technology Sydney
Sydney, Australia
barry.drake@uts.edu.au

Abstract

This paper introduces a novel method for generating differentially private synthetic datasets that harnesses Bayesian networks to ensure the preservation of essential statistical properties and referential integrity across linked tables. To address the dual challenges of maintaining privacy and minimizing computational overhead, we introduce a decomposition scheme for additive Laplacian noise that significantly reduces computational costs while enhancing the efficiency of the differential privacy framework. Our methodology offers a robust solution for creating synthetic datasets that not only mimic the statistical characteristics of original datasets, but also safeguard sensitive information against inference attacks. Through comprehensive evaluations, we demonstrate the practicality and effectiveness of our approach, which achieves a significant speedup in noise injection, thereby facilitating real-time data analysis. This breakthrough contributes to the broader accessibility of complex data analysis, particularly benefiting sectors dealing with sensitive information by improving data privacy and security measures. Our findings represent a significant advancement in statistical methodologies and software, underscoring the ongoing necessity for innovation in data processing techniques.

CCS Concepts

• **Computing methodologies** → **Probabilistic reasoning**; • **Security and privacy** → **Privacy-preserving protocols**; Data anonymization and sanitization.

Keywords

Synthetic Dataset, Differential Privacy, Bayesian Networks

ACM Reference Format:
Sandeep Suresh, Guangsheng Zhang, Bo Liu, and Barry Drake. 2025. Cost-Efficient and Privacy-Preserving Synthesis of Complex Sensitive Data. In *2025 Australasian Computer Science Week (ACSW 2025), February 10–13, 2025, Brisbane, QLD, Australia.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3727166.3727170

ACSW 2025, Brisbane, QLD, Australia
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-1507-5/25/02
https://doi.org/10.1145/3727166.3727170

1 Introduction

The creation and utilization of synthetic datasets have emerged as a pivotal method in current data science and privacy preservation. A synthetic dataset consists of artificially generated records designed to encapsulate the essential properties of a reference dataset, thereby mimicking its statistical characteristics. Synthetic data generation has applications in various domains, particularly as a surrogate for highly sensitive datasets (e.g. healthcare [8]). These synthetic datasets serve a multitude of purposes, from pedagogical demonstrations and testing to marketing applications; safeguarding the confidentiality of the original data. This strategic utility is reflected in recent academic works [2], [18], [10], highlighting the increasing importance of synthetic data sets in data-driven research and industry applications.

Most existing synthetic dataset generation methods either use generative adversarial networks (GANs) or probabilistic graphic models. GAN is a class of machine learning models designed to generate data, often in the form of images, audio, or text. Specifically, medGAN [3] and its successor medBGAN [1] are used to synthesize health data. However, they are limited to binary and count variables, which represent a small subset of potential medical data types. On the other hand, Bayesian networks (BN), one of the representative types of probabilistic graphical models, have been widely used to generate synthetic data, especially tabular data [19]. Bernstein et al. [2] explain much of the theory underlying the modelling approach. Tucker et al. [18] provides a comprehensive review of approaches to synthetic data generation. They chose a generative probabilistic modelling approach to synthetic Clinical Practice Research Datalink (CPRD) because it allowed combining machine learning with expert knowledge. Kaur et al. [10] proposed a system for generating synthetic data using structure learning with BNs. They demonstrate that BNs perform better than GANs across nine separate evaluation measures. Furthermore, it argues that BNs are more transparent and humanly understandable than GANs.

However, while synthetic datasets offer preliminary privacy protection, they are not inherently immune to inference attacks. This vulnerability is illuminated in a study by Stadler et al. [16], emphasizing the need for additional privacy protection methods, with differential privacy (DP) being a prominent choice. BNs work well with DP applications, providing an accessible framework for querying statistical properties and setting parameters based on conditional probabilities. These networks make it easier to calculate probabilities straight from the data, avoiding the need to deal with events that never happen in the real data.

Yet, the implementation of DP poses significant computational challenges, particularly when we need to add random noise to cross-tabs (a cross-tab is the number of occurrences of different combinations of values of the random variables) to mask individual data contributions. The main problem lies in handling zero-frequency elements: while it is easy to hide data that occurs with random noise, data that never happens (showing some data combinations do not exist) needs careful handling to keep DP safe. This is not merely a theoretical concern but a practical challenge, as the computational cost of noise addition scales with the number of possible data combinations. In large datasets, there can be so many combinations that the time and resources needed to handle them become too much. To overcome this, our paper proposes a novel and cost-effective approach to generating differentially private synthetic datasets.

The core of our methodology is the construction of a Bayesian network, which serves as the generative model for synthetic data. We incorporate additional privacy-preserving measures to provide robust differential privacy guarantees for the generated datasets. To alleviate the computational burden associated with noise addition, we introduce a decomposition scheme for additive Laplacian noise, significantly reducing computational costs. Our main contributions can be summarized as follows:

- We propose a novel approach for generating differentially private synthetic datasets using BNs and a sampling technique with referential integrity. Users can efficiently query the generative model to gather synthetic datasets and ensure the preservation of privacy.
- We propose a decomposition scheme for additive Laplacian noise, reducing computational costs and enhancing efficiency in the differential privacy framework.
- We evaluate the proposed noise generator in terms of privacy gain, accuracy and time taken to generate the noise. The generated synthetic data is also evaluated to determine its privacy performance.

The remainder of the paper is organized as follows. Section 2 presents the preliminaries. Our proposed method is detailed in Section 3. Section 4 evaluates the performance and practical implications of our method. And Section 5 concludes the work.

2 Preliminaries

2.1 Bayesian Networks

A Bayesian network (BN) defines conditional independency between random variables by its structure, which is a directed acyclic graph (DAG) with random variables as graph nodes. Each node represents a random variable, and the nodes are connected to each other by directed arcs. The model structure implies that each random variable will have a set of zero or more parent variables. A random variable and its parents are collectively known as a family. Thus, a BN of n random variables has n families, each family has one child variable and zero or more parent variables.

Each family of a BN is associated with a potential function which gives the conditional probability over the values of the child variable, conditioned on values of the parent variables. If all the random variables of the BN are discrete, then the potential function may be represented as a table, known as a conditional probability table or CPT. Note that the potential function need not be a CPT for other graphical models. Mathematically, it is convenient to think of each potential function as a factor from the combined states of the family to a real number representing the conditional probability.

To fully define a BN, one needs to define three things: (1) the random variables (each with their possible values), (2) the network structure, and (3) the values for all network parameters (i.e. the values of all CPTs).

Parameter values can be defined from data using maximum likelihood learning. The values for each parameter can be determined from a cross-table of each family. That is, for each possible combination of values of the family, a cross-table counts the number of rows in the reference data that match the combination of values. Each cross-table defines an empirical partial joint probability distribution over its random variables.

2.2 Synthetic Data Generation Using Bayesian Networks

At a top level, there are two approaches to generate synthetic data [18]. One is taking records from a reference database and perturbing them, e.g., changing values and mixing up record pieces. The other approach is to construct a generating process that emits data, where the generating process is designed to somehow match the reference database. For both approaches, it is difficult to construct a perturbation process that guarantees both privacy and statistical similarity to the reference data. Choosing the right kinds of perturbation to protect privacy yet maintain similarity is a challenging trade-off that requires careful implementation.

In addition, no generating process can provide information beyond what is inherent in the process. This means that privacy assurances can focus on the generating process rather than the resulting data. That focus is particularly advantageous when the generating process is based on generative models as privacy assurance and then may focus on the information embedded in the models. Furthermore, a generation process is potentially unlimited in the data it may produce, without compromising previously assured privacy guarantees.

Considering these things, we were motivated to use the existing databases as references to create generative models. We us Bayesian Networks (BN) [11] because a BN can be directly examined to determine the statistical properties of the data they generate. Our proposed method takes the BN structure as input. For example, the structure may be defined manually. Values for the parameters are set algorithmically as described in the sections below.

There are some advantages to use BN. First, BN is computationally efficient and scales well with the dimensionality of the dataset. Second, the directed acyclic graph can also be utilized for exploring the causal relationships across the variables. With these advantages, there is one disadvantage. Even though the full joint distribution's factorization [7] is general enough to include any possible dependency structure, in practice, simplifying assumptions on the graphical structure is made to ease model inference. Such simplifications may neglect to represent higher-order dependencies.

2.3 Sampling

Once the generative model is constructed, synthetic data may be generated by sampling the joint probability implied by a BN model.

As a BN defines a joint probability distribution, that distribution may be sampled to provide potentially unlimited records. In our proposed method, we compile the BN into an efficient intermediate representation [4, 15]. This permits efficient conditioning and marginalisation of the distribution; thus a subset of random variables may be sampled conditioned on the state of others. Importantly, we use inverse transform sampling [5] which enables individual samples to be drawn from the model without requiring burn-in as is common for Monte-Carlo Markov Chain sampling [9].

We acknowledge that the BN-based method may be complex compared to directly perturbing the reference datasets, both from an implementation perspective and computational resources. However, the complexity of the implementation can be managed by relying on existing software [12, 13, 19]. The computational complexity for calibrating and sampling models can be managed by compiling a BN to efficient arithmetic circuits.

3 Privacy-preserving Synthetic Data Generation

3.1 Synthetic Data Generator Model

Our proposed synthetic data generator takes a probabilistic modelling approach to generate synthetic data. As shown in Figure. 1, the whole process consists of two main steps: (1) create the model; (2) sample the model.

We will explain the details in the rest of this subsection.

3.1.1 *Create Model.*

The core of this approach is a generative model. The modelling formalism used to develop the proposed model is Probabilistic Graphical Models [11]. A probabilistic graphical model defines a joint probability distribution over a set of random variables. Random variables are represented by circles.

Directed arcs between random variables indicate statistical dependencies between the random variables of the parents and the child (a family). The random variables and arcs define the model structure. They form a 'directed acyclic graph'. It may have cycles, but not directed cycles. Each family has a table of parameter values, which can be derived from a cross table that covers the random variables.

3.1.2 *Sample Model.*

Since the above-generated model defines a joint probability distribution over a set of random variables, it can be used to generate samples such that in the limit (of a large number of samples) the joint probability distribution of the samples approaches defined by the model.

Our proposed model allows sampling from a conditional probability distribution and a marginal joint probability distribution. A conditional probability of X given Y is written as $P(X|Y)$. Intuitively, this is the probability of X assuming Y is true. Formally it is defined as

$$P(X\&Y) = P(X|Y) \cdot P(Y), \tag{1}$$

A marginal joint probability distribution samples from a subset of the random variables of a model, while ignoring the others. Formally, if the random variables of a model are split into two, X and Y, then the joint probability of X ignoring Y is defined as

$$P(X) = \sum_{y} P(X\&(Y = y)), \tag{2}$$

where y ranges over all possible combinations of values for Y.

Table 1: Example of cross-tab.

A	B	C	weight
yes	yes	yes	10
yes	yes	no	6
yes	no	yes	1
yes	no	no	14
no	yes	yes	2
no	yes	no	5
no	no	yes	1
no	no	no	3

Together, the ability to sample a conditional probability distribution and a marginal joint probability distribution allows a single model to represent multiple entities. This sampling process can be repetitively done until we have the required number of data samples.

3.2 Privacy Preservation Schemes

Privacy leakage happens when information can be used to identify an individual. An individual is 'identified' when a link can be established between the information and the individual. Two key steps are required to effectively de-identify personal information: (1) removal or alteration of all personal identifiers from the information, as has been done for many of the publicly available datasets; and (2) removal or alteration of any other information that may, alone or in combination with other information, allow an individual to be reasonably identifiable.

In the process of synthetic dataset generation, we have implemented privacy protection measures during the process to ensure that the privacy leakage is controllable, as well as complying with the regulations.

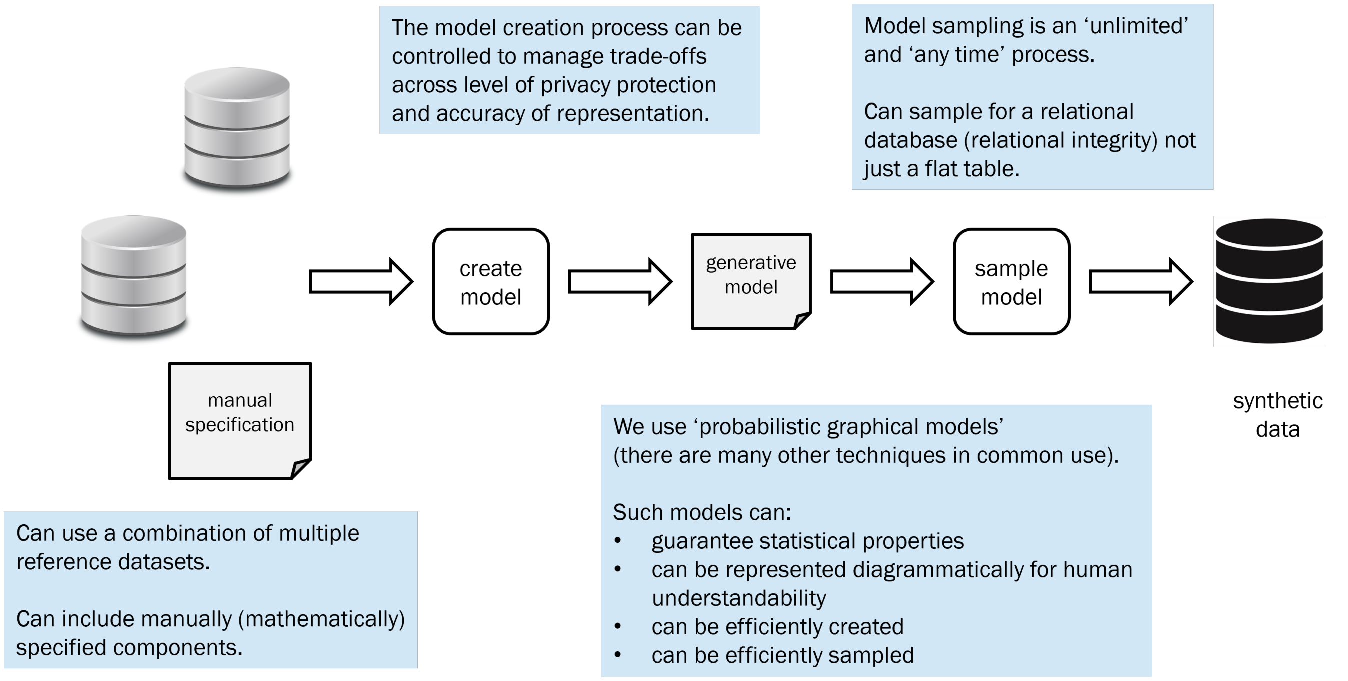
Here, we introduce a scheme called ***cross-tab***. A cross-tab for a collection of random variables provides a count of a number of occurrences of different combinations of values of the random variables. For example, given random variables A, B and C each with possible values 'yes' and 'no', there are eight different combinations of values.

We generalise the definition of a cross-tab replacing counts with weights, where a weight can take on a non-integer, non-negative value. Table 1 shows an example cross-tab.

Before using each cross-table to define a conditional probability table (CPT), we first include two additional privacy preservation steps: (1) Differential Privacy, i.e., adding Laplacian noise; (2) Suppression via defining minimum cell size. These are shown diagrammatically in Figure 2 using a simple fictitious example.

3.2.1 *Differential Privacy.*

In the first step, we add random Laplacian noise values to every cross-table count. The amount of noise is controlled by a parameter called noise scale. A noise scale value



The model creation process can be controlled to manage trade-offs across level of privacy protection and accuracy of representation.

Model sampling is an 'unlimited' and 'any time' process.

Can sample for a relational database (relational integrity) not just a flat table.

Can use a combination of multiple reference datasets.

Can include manually (mathematically) specified components.

We use 'probabilistic graphical models' (there are many other techniques in common use).

Such models can:

- guarantee statistical properties
- can be represented diagrammatically for human understandability
- can be efficiently created
- can be efficiently sampled

Figure 1: Diagram of our privacy-preserving synthetic data generation

of x means that random noise with zero mean and σ variance is added or subtracted from each count. However, the actual value added or subtracted is a random floating-point number. If this step causes a count to go negative, the count is reset to zero.

In a differential privacy process, we perturb cross-tab weights by adding noise to weights and suppressing weights below a threshold. In our case, noise is drawn from a Laplacian distribution, $Lap(\mu, b)$, with a Probability Density Function (PDF):

$$p(x|\mu, b) = \frac{1}{2b} \exp\left(-\frac{|x-\mu|}{b}\right), \tag{3}$$

and the corresponding Cumulative Distribution Function (CDF) is:

$$P(X < x|\mu, b) = c_{\mu,b}(x) = \frac{1}{2} + \frac{\operatorname{sgn}(x-\mu)}{2} \exp\left(\frac{x-\mu}{b}\right). \tag{4}$$

In particular, we set $\mu = 0$ and $b = \Delta f / \epsilon$ where ϵ and Δf are differential privacy parameters known as epsilon and sensitivity, respectively.

3.2.2 Suppression. In the second step, we enforce minimum cell size, y. That is, if any count is less than y then the count is reset to zero. Minimum cell size suppression can be performed as part of arbitrary post-processing, the likes of which are inherently protected, but that has no bearing on the Differential Privacy property.

The suppression of weights is controlled by a parameter known as min-cell-size which we represent by r. Cross-tab rows with weight $< r$ are not made available to subsequent processes. In this case, subsequent processes may assume a weight for unavailable combinations of values of the random variables, for example, 0, 1, or $r/2$, etc. We consider only where r is positive and rounding of values is not used. (Note that any rounding system may be translated to a non-rounding system by subtracting 0.5 from r.)

3.3 Reducing Computational Cost via Decomposition

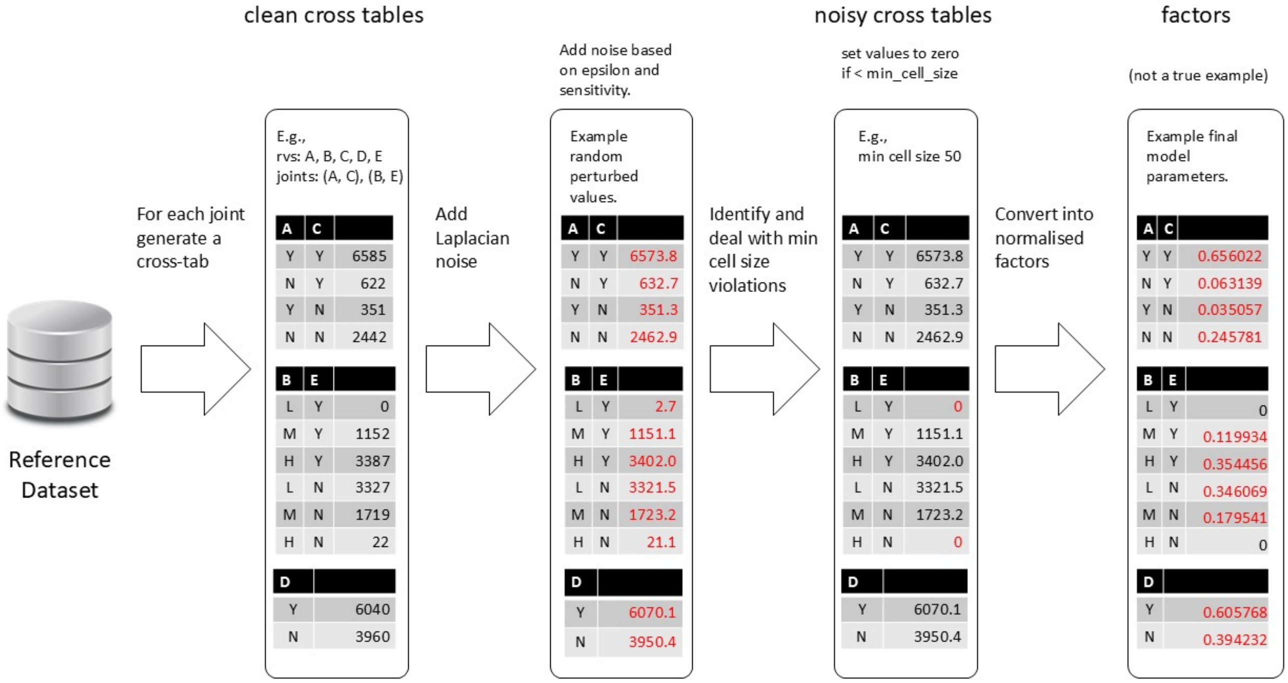
In some practical situations, when a cross-tab is created, rows with zero values are not reported. For example, when using SQL to query a database, "select A, B, C, count(*) from τ group by A, B, C", zero weights are not returned. This provides a significant computational advantage when the space of possible values is large and many of the weights are zero. It provides efficiency for the creation and storage of cross-tabs. It may also lead to efficient models that use such cross-tabs.

However, to achieve strict ϵ-Differential Privacy, it may require constructing the full state space of possible values (i.e. including zeros). This poses a dilemma when adding noise to a cross-tab. Noise must be added to both the zero and non-zero weights of a cross-tab which forces the noise process to consider all possible combinations of values, even if they are not explicitly represented in the cross-tab. This can lead to unacceptable computational costs.

We note that when a min-cell-size is enforced subsequent to adding noise, then the combination of adding noise, then suppressing low weights may lead to no overall effect on a cross-table row. This inspires an approach that decomposes the perturbation process in a more efficient manner.

As shown in Table 2, the perturbation of a weight conforms to one of four cases:

We propose restructuring the perturbation process so that case ZS involves no processing to provide a computational advantage.


clean cross tables
noisy cross tables
factors
Reference Dataset
For each joint generate a cross-tab
Add Laplacian noise
Add noise based on epsilon and sensitivity.
Identify and deal with min cell size violations
set values to zero if < min_cell_size
Convert into normalised factors
(not a true example)
E.g., rvs: A, B, C, D, E joints: (A, C), (B, E)
Example random perturbed values.
E.g., min cell size 50
Example final model parameters.

A	C	
Y	Y	6585
N	Y	622
Y	N	351
N	N	2442

B	E	
L	Y	0
M	Y	1152
H	Y	3387
L	N	3327
M	N	1719
H	N	22

D	
Y	6040
N	3960

A	C	
Y	Y	6573.8
N	Y	632.7
Y	N	351.3
N	N	2462.9

B	E	
L	Y	2.7
M	Y	1151.1
H	Y	3402.0
L	N	3321.5
M	N	1723.2
H	N	21.1

D	
Y	6070.1
N	3950.4

A	C	
Y	Y	6573.8
N	Y	632.7
Y	N	351.3
N	N	2462.9

B	E	
L	Y	0
M	Y	1151.1
H	Y	3402.0
L	N	3321.5
M	N	1723.2
H	N	0

D	
Y	6070.1
N	3950.4

A	C	
Y	Y	0.656022
N	Y	0.063139
Y	N	0.035057
N	N	0.245781

B	E	
L	Y	0
M	Y	0.119934
H	Y	0.354456
L	N	0.346069
M	N	0.179541
H	N	0

D	
Y	0.605768
N	0.394232

Figure 2: A demonstration of the noise injection processes for privacy preservation.

Table 2: Four different cases of the combination of initial and subsequent weight.

Case Name	Initial Weight	Subsequent Weight After Suppression
ZS	Z: weight = zero (not represented in the cross-tab)	S: addition of noise leaves weight < r (suppressed)
ZN	Z: weight = zero (not represented in the cross-tab)	N: addition of noise leaves weight ≥ r (not suppressed)
WS	W: weight > zero (represented in the cross-tab)	S: addition of noise leaves weight < r (suppressed)
WN	W: weight > zero (represented in the cross-tab)	N: addition of noise leaves weight ≥ r (not suppressed)

Observe that the W cases (WS and WN) do not need special treatment, only the Z cases. We treat the Z cases collectively as a repeated application of noise, then a suppression decision.

For an individual row, let the probability of case ZN given Z be α, thus:

$$\begin{aligned} \alpha &= P(ZN|Z) && (5)\\ &= \int_r^\infty \frac{1}{2b}\exp\left(-\frac{x}{b}\right)dx && (6)\\ &= \frac{1}{2}\exp\left(-\frac{r}{b}\right). && (7)\end{aligned}$$

Consider a cross-tab of m rows with zero weight. The number of ZN cases is a binomial random variable with distribution $B(m,\alpha)$.

The noise distribution in the case ZN is a ‘truncated’ Laplacian. Noting that $\mu = 0$ and $r > 0$ we have PDF

$$p(x|r,b,ZN) = \begin{cases} \frac{1}{2b}\exp\left(-\frac{x}{b}\right) & \text{if } x > r\\ 0 & \text{if } x \le r \end{cases}, \qquad (8)$$

where z is the required normalisation constant. The corresponding CDF is:

$$P(X < x|r,b,ZN) = \frac{1}{z}\begin{cases} \int_r^\infty \exp\left(-\frac{x}{b}\right) & \text{if } x > r\\ 0 & \text{if } x \le r \end{cases} \qquad (9)$$

$$= \frac{1}{2z}\begin{cases} \exp\left(-\frac{x}{b}\right) & \text{if } x > r\\ 0 & \text{if } x \le r \end{cases}, \qquad (10)$$

with

$$\frac{1}{z} = \int_r^\infty \exp\left(-\frac{x}{b}\right) = \frac{1}{2}\exp\left(-\frac{x}{b}\right). \tag{11}$$

Therefore,

$$P(X < x|r, b, ZN) = \begin{cases} \exp\left(-\frac{r-x}{b}\right) & \text{if } x > r \\ 0 & \text{if } x \le r \end{cases}. \tag{12}$$

The inverse CDF is $r - b \ln x$. Simple inverse transform sampling may used to draw samples from this distribution.

Putting this all together, Algorithm 1 is a process to add noise and suppression for a cross-tab, with m rows of zero weight.

Algorithm 1: Privacy-preserving Cross-tab Generation for Case ZN.

```
   Data: Parameters $r$ and $b$, and no. of zero weight rows $m$.
   Result: Privacy preserving Cross-tab variables $w$.
1  Set $\alpha = \frac{1}{2}\exp\left(-\frac{r}{b}\right)$;
2  Draw a random variate $n \sim B(m, \alpha)$;
3  Randomly choose $n$ different combinations of values for the
     cross-tab random variables, where each chosen
     combination was not a combination with row weight $> 0$;
4  for each chosen combination $c$ do
5      Draw a random variate $x \sim U(0, 1)$;
6      Let $w = r - b\ln(x)$;
7      Include a row in the cross-tab with combination $c$ and
         weight $w$;
8  for each row with original weight $w > 0$ do
9      Draw a random variate $x \sim Lap(0, b)$;
10     if $w + x < r$ then
11         Suppress the row: $w = 0$;
12     else
13         Set the row weight to $w = w + x$;
```

Line 3 of the algorithm is the challenging step in this process. If $n \ll m$ then this step may be achieved using rejection sampling, i.e., by uniformly sampling possible combinations and keeping a sample only if it is not already in the cross-tab. To be computationally efficient, m should be large, and α should be small, i.e., a large value for r relative to b.

4 Performance Evaluation

4.1 Experimental Settings

4.1.1 Dataset. We use the Texas dataset [17] for performance evaluation. The Texas Hospital Discharge dataset is a large public-use data file provided by the Texas Department of State Health Services. The dataset we use consists of 50,000 records uniformly sampled from a pre-processed data file that contains patient records from the year 2013. We retain 18 data attributes, of which 11 are categorical and 7 continuous.

4.2 Evaluation Metrics

Performance evaluation can be conducted in three aspects: utility, privacy and computational cost (scalability).

1. **Utility:** We use three key approaches to evaluating the utility of synthetic data, as suggested in [6].
   - **Structural Similarity**: confirm variable names, types, formats and 'edit check' rules are satisfied.
   - **Bias and Stability**: measure the variation across each time synthetic data is generated or the generating model is calibrated.
   - **Statistical Similarity**: measure the statistical differences between real and synthetic datasets. Histogram Intersection (HI) and Kullback–Leibler Divergence (KL) are two different methods used to measure the similarity between two probability distributions or histograms. The HI between two histograms A and B is calculated as $HI(A, B) = \sum_i \min(A(i), B(i))$, where $A(i)$ and $B(i)$ are the values of the i^{th} bins in histograms A and B, respectively. A higher HI value indicates a greater similarity. On the other hand, The KL divergence of a distribution Q from a distribution P over the same probability space is defined as $KL(P \parallel Q) = \sum_i P(i) \log\left(\frac{P(i)}{Q(i)}\right)$, where $P(i)$ and $Q(i)$ are the probabilities associated with the i^{th} event in the distributions P and Q, respectively. KL with a value of 0 indicates that the two distributions are identical.
2. **Privacy:** Apart from the theoretical DP guarantee, we also use another privacy metric from Stadler et al. [16].
   - **Privacy gain (PG) [16]**: the privacy gain of publishing a synthetic dataset S in place of the raw data R for target record r_t as the reduction in the adversary's advantage when given access to S instead of R. I.e., $PG \triangleq Adv(R, r_t) - AdV(S, r_t)$. The privacy gain can assess whether synthetic data is, as promised, an effective anonymisation mechanism.
3. **Computational cost:** Compare the computation costs (CPU time or the number of calculations) of adding noise to all cross-tab values and to sparse cross-tabs by decomposing the noise distribution.

4.3 Experimental Results

The probabilistic model representing the Texas dataset is illustrated in Figure 3. Each grey dashed random variable was functionally derived through a process of histogram normalization, involving the categorization of corresponding original random variable values into five distinct buckets. A noteworthy aspect of this model is that the crosstabs generated between these paired variables are not directly extracted from the dataset; hence, they are not inherently subject to noise. Consequently, we introduced noise to a total of 12 such crosstabs. This noise induction employed two distinct approaches: our proposed method (decomposed Laplacian noise) and the standard method (conventional full state space representation).

4.3.1 Utility.

Structural Similarity: Correct variable names, types, and formats are guaranteed to be correct. This is because possible values

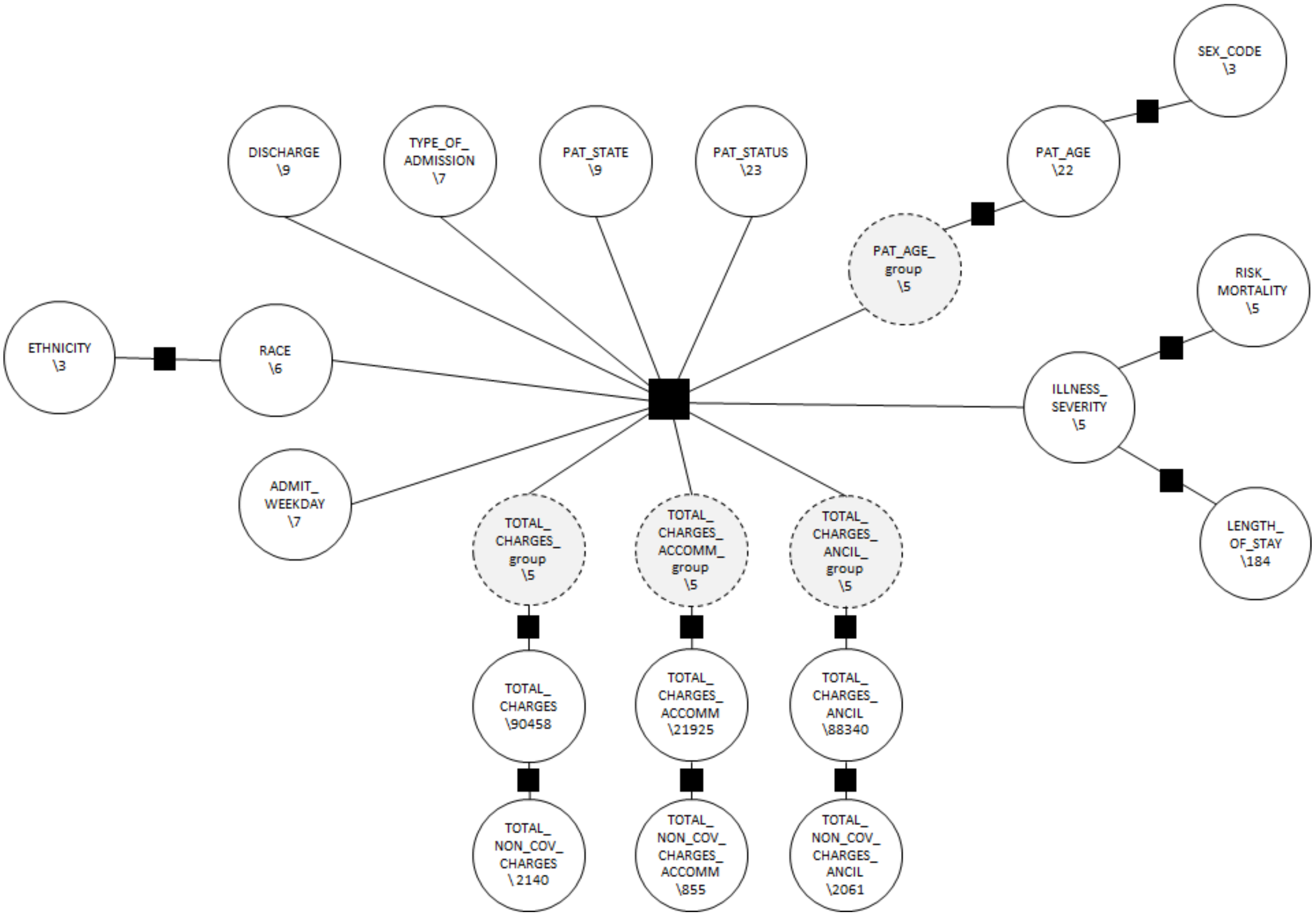


Figure 3: The probabilistic graphical model used for the Texas dataset. Random variables are shown as circles. Crosstabs are shown as squares.

are taken directly from those available in the reference dataset. Edit check rules are not enforced, as they are not known for the Texas dataset. However, any invalid combination of variable values can easily be enforced by ensuring a zero value in a crosstab with that combination.

Bias and Stability: Once a model is created, it fully defines a stationary joint distribution. Therefore, multiple synthetic datasets from the same model exhibit no bias and perfect stability.

Statistical Similarity: The histogram-intersection values were calculated for all the crosstabs in Figure 3 for both the proposed method and the standard method. Figure 4 shows the highest, lowest and median HI values out of all the HI values from the 12 crosstabs. We can see that there is no statistically significant difference between standard and proposed methods. Therefore the proposed method works as well as the standard method in terms of accuracy.

4.3.2 *Privacy.*

Apart from the privacy guarantee provided by DP, we also evaluate the privacy performance using the PG metric via conducting linkability games following [16]. A linkability game involves a privacy challenge between an adversary and a challenger, where the challenger releases a synthetic dataset accessible to the adversary. The adversary's objective is to infer whether a target record originates from the original dataset, leveraging the synthetic dataset and prior knowledge. In addition, the adversary can use different feature extraction techniques to distinguish the feature vectors extracted from synthetic datasets with and without a target record. We provide the results for the generation model using three different feature sets:

- Naive: This feature set is designed to capture the basic statistical properties of the dataset. For each numerical attribute, it calculates the mean, median, and variance. In the case of categorical attributes, it records the number of distinct categories as well as identifies the most and least frequent categories. This approach provides a straightforward summary of the data's central tendency, variability, and category frequencies.
- Histograms: This set focuses on understanding the distribution of data attributes. For numerical attributes, it segments

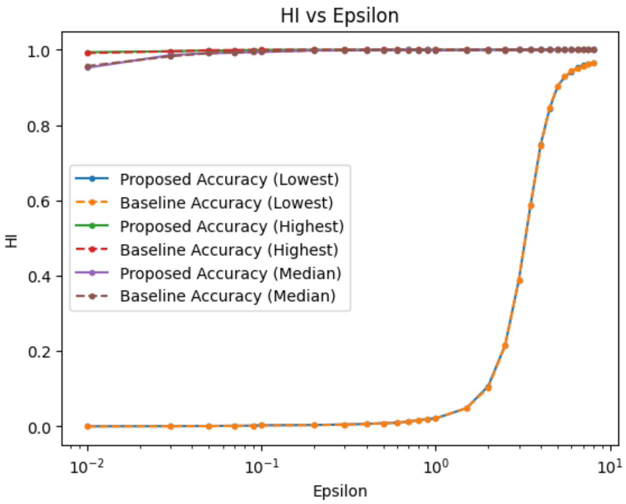
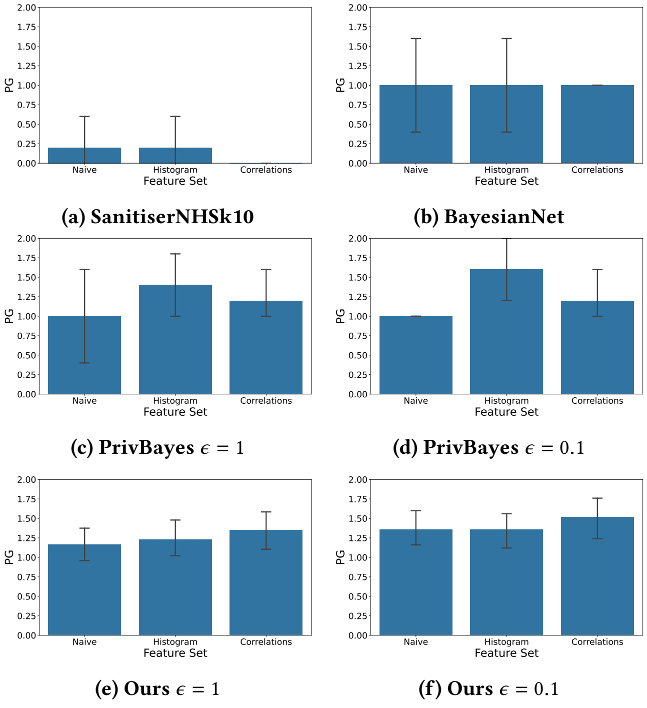


Figure 4: Graph showing the accuracy of the proposed and standard method. The highest, lowest and median histogram-intersection (HI) values are shown.



(a) SanitiserNHSk10

(b) BayesianNet

(c) PrivBayes $\epsilon = 1$

(d) PrivBayes $\epsilon = 0.1$

(e) Ours $\epsilon = 1$

(f) Ours $\epsilon = 0.1$

Figure 5: The comparison of the privacy gain (PG) results of linkability game between several generation models and our method.

data into bins of a configurable size, creating a histogram that reflects the data's distribution. Categorical attributes are treated by counting the frequency of each category. The number of bins for numerical attributes can be adjusted for each dataset, allowing for tailored analysis of data distribution.

- Correlations: This feature set aims to uncover relationships between pairs of attributes within the dataset. It calculates the correlations between numerical attributes directly. For categorical attributes, it first converts them into a numerical format through dummy encoding, which transforms categorical data into a series of binary variables. This enables the computation of a pairwise correlation matrix, offering insights into how attributes relate to each other.

Baseline methods. We use three other synthetic data generation as our baseline methods, namely SanitiserNHSk10, BayesianNet, and PrivBayes.

- SanitiserNHSk10 [14]. This sanitization procedure, described by NHS England, employs a deterministic function to apply a series of predefined row-level transformations to input data (R), producing a sanitized dataset (S) that meets a heuristic privacy definition. Typical transformations include generalization, perturbation, or the deletion of individual rows to enhance data privacy.
- BayesianNet [19]. This method leverages Bayesian Networks (BN) to construct models, yet it lacks integration of privacy preserving mechanisms. It focuses on model accuracy and inference capabilities without addressing the privacy concerns associated with the data it processes.
- PrivBayes [19]. PrivBayes is a differentially private BN model. A synthetic dataset can be sampled from the trained model without any additional privacy budget cost. The primary distinction from our proposed method lies in its approach to differential privacy: during the model training phase, differential privacy noise is introduced to the probabilities of conditional distributions. However, this method does not offer formal Differential Privacy (DP) guarantees, marking a significant difference.

Figure 5 provides the privacy gain (PG) results across different generation models in a linkability game framework. The figure presents various approaches, including SanitiserNHSk10, BayesianNet, and PrivBayes with $\epsilon = 1$ and 0.1, and our proposed method with $\epsilon = 1$ and 0.1. Our method shows consistently robust privacy gains across all feature sets. SanitiserNHSk10 exhibits significantly lower privacy gains across all feature sets, while BayesianNet and PrivBayes show higher privacy gains similar to our method. The privacy parameter ϵ appears to influence the stability of results, with $\epsilon = 0.1$ generally showing better performance than the results for $\epsilon = 1$.

4.3.3 *Computational Cost.*

The time taken to generate noise for all 12 crosstabs were calculated for both the proposed and standard method. The wall clock time was measured for epsilon values ranging from 0.01 to 8. Figure 6 shows the time taken to generate noise for the whole range of epsilon values.

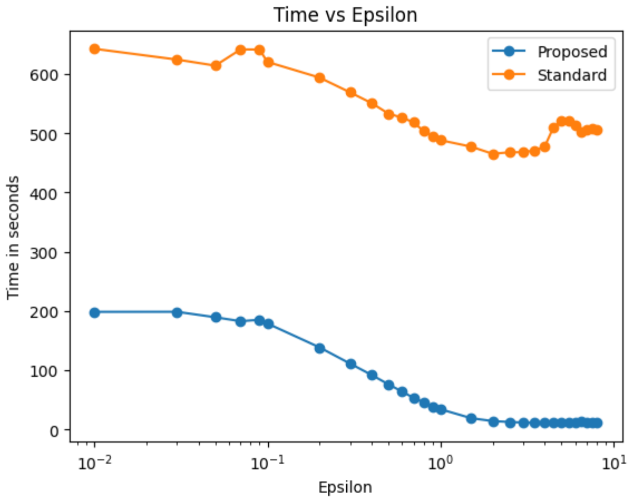


Figure 6: Total time taken for all crosstabs to generate noise for a range of epsilon values.

4.4 Discussion

Although, the proposed method was demonstrated here using Bayesian Networks, the method can work for any statistical model that works on counts. This is because the noise addition process does not rely on the structural assumptions of Bayesian Networks. On the other hand, state-of-the-art synthetic data generators like PrivBayes [19] rely more on Bayesian Networks. They rely on the ability of Bayesian Networks to to decompose a high-dimensional data distribution into a collection of low-dimensional conditional distributions.

In PrivBayes [19], the Bayesian Network is constructed using a differentially private algorithm (using the exponential mechanism for adding noise), that selects attribute-parent pairs based on a modified mutual information score. The relevant differentially private distributions of the data are then computed in the sub-spaces of the Bayesian Network via the Laplace mechanism. The contribution reported for PrivBayes [19] is the application of noise partial joint probabilities rather than the full joint probability distribution. Our contribution builds on this to demonstrate that decomposing the noise process yields additional efficiency gains. So PrivBayes needs Bayesian networks to achieve differential privacy, and it isn't a generic framework that can be directly applied with any statistical model. However, as demonstrated, our proposal can work for any statistical model (that relies on counts) and at the same time provide privacy guarantees in datasets with complex dependencies.

In our proposed method, the min-cell-size parameter practically induces further noise in the model, beyond that added by the Laplacian process. In particular, when a cross table has low counts then the noise from enforcing min-cell-size will be more noticeable. However, low count combinations are exactly the states DP should protect. Low counts may be caused by rare combinations of states or a cross table that has many random variables.

5 Conclusion

The time taken to generate noise for our proposed method and the standard method was calculated over a range of epsilon values. The time taken by the standard method was approximately three times that of the proposed method. This demonstrates the significant computational cost advantage that can be gained by enforcing the min-cell-size and by restructuring the perturbation process for the zero weight cases.

The accuracy of our proposed method was compared against the standard method for generating noise. Histogram-Intersection (HI) was calculated for both methods over a range of epsilon values for all crosstabs. There is no statistically significant difference in accuracy between our proposed method and the standard method. We have also evaluated the privacy performance via the PG metric by conducting linkability games with three baseline data generators. Our approach not only meets Differential Privacy (DP) standards but also enhances overall privacy by reducing potential leakage, providing strong protection for sensitive information in complex, high-dimensional datasets.

Acknowledgments

OpenAI's ChatGPT-4 has been used to proofread and improve the quality of writing of the paper. The authors accept responsibility for the veracity and correctness of this paper.

References

[1] M. K. Baowaly, C. C. Lin, C. L. Liu, and K. T. Chen. 2019. Synthesizing Electronic Health Records using Improved Generative Adversarial Networks. *Journal of the American Medical Informatics Association* 26, 3 (2019), 228–241. https://doi.org/10.1093/jamia/ocy142

[2] Garrett Bernstein, Ryan McKenna, Tao Sun, Daniel Sheldon, Michael Hay, and Gerome Miklau. 2017. Differentially private learning of undirected graphical models using collective graphical models. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70* (Sydney, NSW, Australia) *(ICML'17)*. JMLR.org, 478–487.

[3] Edward Choi, Siddharth Biswal, Bradley Malin, Jon Duke, Walter F. Stewart, and Jimeng Sun. 2017. Generating Multi-label Discrete Patient Records using Generative Adversarial Networks. In *Proceedings of the 2nd Machine Learning for Healthcare Conference (Proceedings of Machine Learning Research, Vol. 68)*, Finale Doshi-Velez, Jim Fackler, David Kale, Rajesh Ranganath, Byron Wallace, and Jenna Wiens (Eds.). PMLR, 286–305. https://proceedings.mlr.press/v68/choi17a.html

[4] A. Darwiche. 2009. *Modeling and Reasoning with Bayesian Networks.* Cambridge University Press. https://doi.org/https://doi.org/10.1017/CBO9780511811357

[5] L. Devroye. 1986. *Non-Uniform Random Variate Generation.* Springer New York. https://books.google.com.au/books?id=mEw_AQAAIAAJ

[6] K. Emam. 2020. Seven Ways to Evaluate the Utility of Synthetic Data. *IEEE Security & Privacy* 18, 4 (2020), 56–59. https://doi.org/10.1109/msec.2020.2992821

[7] A. Goncalves, P. Ray, B. Soper, J. Stevens, L. Coyle, and A. P. Sales. 2020. Generation and Evaluation of Synthetic Patient Data. *BMC Medical Research Methodology* 20, 1 (2020), 1–40. https://doi.org/10.1186/S12874-020-00977-1/TABLES/17

[8] NSW Health. [n. d.]. Lumos Dataset. https://www.health.nsw.gov.au/lumos. Accessed: 2024-11-15.

[9] G. L. Jones and J. P. Hubert. 2004. Sufficient Burn-In for Gibbs Samplers for a Hierarchical Random Effects Model. *The Annals of Statistics* 32, 2 (2004), 784–817. https://doi.org/10.1214/009053604000000184

[10] D. Kaur, M. Sobiesk, S. Patil, J. Liu, P. Bhagat, A. Gupta, and N. Markuzon. 2021. Application of Bayesian Networks to Generate Synthetic Health Data. *Journal of the American Medical Informatics Association: JAMIA* 28, 4 (2021), 801–811. https://doi.org/10.1093/jamia/ocaa303

[11] Daphne Koller and Nir Friedman. 2009. *Probabilistic Graphical Models: Principles and Techniques - Adaptive Computation and Machine Learning.* The MIT Press.

[12] Sofiane Mahiou, Kai Xu, and Georgi Ganev. 2022. dpart: Differentially Private Autoregressive Tabular, a General Framework for Synthetic Data Generation. arXiv:2207.05810 [cs.LG] https://arxiv.org/abs/2207.05810

[13] Ryan McKenna, Gerome Miklau, and Daniel Sheldon. 2021. Winning the NIST Contest: A scalable and general approach to differentially private synthetic data. *ArXiv* abs/2108.04978 (2021), 1. https://api.semanticscholar.org/CorpusID:

236976348

[14] NHS England. 2020. A&E Synthetic Data. https://data.england.nhs.uk/dataset/a-e-synthetic-data. Accessed: 2020-11-12.

[15] Yujia Shen, Arthur Choi, and Adnan Darwiche. 2016. Tractable Operations for Arithmetic Circuits of Probabilistic Models. In *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett (Eds.), Vol. 29. Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2016/file/5a7f963e5e0504740c3a6b10bb6d4fa5-Paper.pdf

[16] Theresa Stadler, Bristena Oprisanu, and Carmela Troncoso. 2022. Synthetic Data – Anonymisation Groundhog Day. In *31st USENIX Security Symposium (USENIX Security 22)*. USENIX Association, Boston, MA, 1451–1468. https://www.usenix.org/conference/usenixsecurity22/presentation/stadler

[17] The Open Data Institute. 2023. *Diagnosing the NHS: SynAE*. The Open Data Institute. Accessed 2023-10-01.

[18] A. Tucker, Z. Wang, Y. Rotalinti, and P. Myles. 2020. Generating High-Fidelity Synthetic Patient Data for Assessing Machine Learning Healthcare Software. *NPJ Digital Medicine* 3, 1 (2020), 147. https://doi.org/10.1038/s41746-020-00353-9

[19] Jun Zhang, Graham Cormode, Cecilia M. Procopiuc, Divesh Srivastava, and Xiaokui Xiao. 2017. PrivBayes: Private Data Release via Bayesian Networks. *ACM Transactions on Database Systems (TODS)* 42, 4 (2017), 25.