Mitigating Statistical Bias within Differentially Private Synthetic Data

Sahra Ghalebikesabi[1] **Harrison Wilde**[2] **Jack Jewson**[3] **Arnaud Doucet**[1]
Sebastian Vollmer[5] **Chris Holmes**[1]

[1]University of Oxford
[2]University of Warwick
[3]Universitat Pompeu Fabra
[5]University of Kaiserslautern, German Research Centre for Artificial Intelligence (DFKI)

Abstract

Increasing interest in privacy-preserving machine learning has led to new and evolved approaches for generating private synthetic data from undisclosed real data. However, mechanisms of privacy preservation can significantly reduce the utility of synthetic data, which in turn impacts downstream tasks such as learning predictive models or inference. We propose several re-weighting strategies using privatised likelihood ratios that not only mitigate statistical bias of downstream estimators but also have general applicability to differentially private generative models. Through large-scale empirical evaluation, we show that private importance weighting provides simple and effective privacy-compliant augmentation for general applications of synthetic data.

1 INTRODUCTION

The prevalence of sensitive datasets, such as electronic health records, contributes to a growing concern for violations of an individual's privacy. In recent years, the notion of Differential Privacy (Dwork et al., 2006) has gained popularity as a privacy metric offering statistical guarantees. This framework bounds how much the likelihood of a randomised algorithm can differ under neighbouring real datasets. We say two datasets $\mathcal{D}$ and $\mathcal{D}'$ are neighbouring when they differ by at most one observation. A randomised algorithm $g : \mathcal{M} \to \mathcal{R}$ satisfies (ϵ, δ)-differential privacy for $\epsilon, \delta \geq 0$ if and only if for all neighbouring datasets $\mathcal{D}, \mathcal{D}'$ and all subsets $S \subseteq \mathcal{R}$, we have

$$\Pr(g(\mathcal{D}) \in S) \leq \delta + e^{\epsilon} \Pr(g(\mathcal{D}') \in S).$$

The parameter ϵ is referred to as the privacy budget; smaller ϵ quantities imply more private algorithms.

Injecting noise into sensitive data according to this paradigm allows for datasets to be published in a private manner. With the rise of generative modelling approaches, such as Generative Adversarial Networks (GANs) (Goodfellow et al., 2014), there has been a surge of literature proposing generative models for differentially private (DP) synthetic data generation and release (Jordon et al., 2019; Xie et al., 2018; Zhang et al., 2017). These generative models often fail to capture the true underlying distribution of the real data, possibly due to flawed parametric assumptions and the injection of noise into their training and release mechanisms. The constraints imposed by privacy-preservation can lead to significant differences between nature's true data generating process (DGP) and the induced synthetic data generating process (SDGP) (Wilde et al., 2020). This increases the bias of estimators trained on data from the SDGP which reduces their utility.

Recent literature has proposed techniques to decrease this bias by modifying the training processes of private algorithms. These approaches are specific to a particular synthetic data generating method (Zhang et al., 2018; Frigerio et al., 2019; Neunhoeffer et al., 2020), or are query-based (Hardt and Rothblum, 2010; Liu et al., 2021) and are thus not generally applicable. Hence, we propose several post-processing approaches that aid mitigating the bias induced by the DP synthetic data.

While there has been extensive research into estimating models directly on protected data without leaking privacy, we argue that releasing DP synthetic data is crucial for rigorous statistical analysis. This makes providing a framework to debias inference on this an important direction of future research that goes beyond the applicability of any particular DP estimator. Because of the post-processing theorem (Dwork et al., 2014), any function on the DP synthetic data is itself DP. This allows deployment of standard statistical analysis tooling that may otherwise be unavailable for DP estimation. These include 1) exploratory data analysis, 2) model verification and analysis of model diagnostics, 3) private release of (newly developed) models for which no

Proceedings of the 38th Conference on Uncertainty in Artificial Intelligence (UAI 2022), PMLR 180:696–705.

DP analogue has been derived, 4) the computation of confidence intervals of downstream estimators through the non-parametric bootstrap, and 5) the public release of a data set to a research community whose individual requests would otherwise overload the data curator. This endeavour could facilitate the release of data on public platforms like the UCI Machine Learning Repository (Lichman, 2013) or the creation of data competitions, fuelling research growth for specific modelling areas.

This motivates our main contributions, namely the formulation of multiple approaches to generating DP importance weights that correct for synthetic data's issues. In particular:

- The bias estimation of an existing DP importance weight estimation method, and the introduction of an unbiased extension with smaller variance (Section 3.3).
- An adjustment to DP Stochastic Gradient Descent's sampling probability and noise injection to facilitate its use in the training of DP-compliant neural network-based classifiers to estimate importance weights from combinations of real and synthetic data (Section 3.4).
- The use of discriminator outputs of DP GANs as importance weights that do not require any additional privacy budget (Section 3.5).
- An application of importance weighting to correcting for the bias incurred in Bayesian posterior belief updating with synthetic data motivated by the results from (Wilde et al., 2020) and to exhibit our methods' wide applicability in frequentist and Bayesian contexts (Section 3.1).

2 BACKGROUND

Before we proceed, we provide some brief background on bias mitigation in non-private synthetic data generation.

2.1 DENSITY RATIOS FOR NON-PRIVATE GANS

Since their introduction, GANs have become a popular tool for synthetic data generation in semi-supervised and unsupervised settings. GANs produce realistic synthetic data by trading off the learning of a generator Ge to produce synthetic observations, with that of a classifier Di learning to correctly classify the training and generated data as real or fake. The generator Ge takes samples from the prior $u \sim p_u$ as an input and generates samples $Ge(u) \in X$. The discriminator Di takes an observation $x \in X$ as input and outputs the probability $Di(x)$ of this observation being drawn from the true DGP. The classification network Di distinguishes between samples from the DGP with label $y = 1$ and distribution p_D, and data from the SDGP with label $y = 0$ and distribution p_G. Following Bayes' rule we can show that the output of $Di(x)$, namely the probabilities $\widehat{p}(y = 1|x)$ and $\widehat{p}(y = 0|x)$, can be used for importance weight estimation:

$$\frac{\widehat{p}_D(x)}{\widehat{p}_G(x)} = \frac{\widehat{p}(x|y = 1)}{\widehat{p}(x|y = 0)} = \frac{\widehat{p}(y = 1|x)}{\widehat{p}(y = 0|x)} \frac{\widehat{p}(y = 0)}{\widehat{p}(y = 1)}. \quad (1)$$

This observation has been exploited in a stream of literature focusing on importance weighting (IW) based sampling approaches for GANs. Grover et al. (2019) analyse how importance weights of the GAN's outputs can lead to performance gains; extensions include their proposed usage in rejection sampling on the GAN's outputs (Azadi et al., 2018), and Metropolis–Hastings sampling from the GAN alongside improvements to the robustness of this sampling via calibration of the discriminator (Turner et al., 2019). To date, no one has leveraged these discriminator-based IW approaches in DP settings where the weights can mitigate the increased bias induced by privatised data models.

2.2 DIFFERENTIAL PRIVACY IN SYNTHETIC DATA GENERATION

Private synthetic data generation through DP GANs is built upon the post processing theorem: If Di is (ϵ, δ)- DP, then any composition $Di \circ Ge$ is also (ϵ, δ)-DP (Dwork et al., 2014) since Ge does not query the protected data. Hence, to train private GANs, we only need to privatise the training of their discriminators, see e.g. Hyland et al. (2018). Xie et al. (2018) propose DPGAN, a Wasserstein GAN which is trained by injecting noise to the gradients of the discriminator's parameters. In contrast, Jordon et al. (2019) privatise the GAN discriminator by using the Private Aggregation of Teacher Ensembles algorithm, leading to a model architecture called PATE-GAN. Recently, Torkzadehmahani et al. (2019) proposed DPCGAN as a conditional variant to DP-GAN that uses an efficient moments accountant. In contrast, PrivBayes (Zhang et al., 2017) learns a DP Bayesian network and does not rely on a GAN-architecture. Other generative approaches, for instance, include Chen et al. (2018); Acs et al. (2018). See Abay et al. (2018); Fan (2020) for an extensive overview of more DP generative approaches.

Differentially private bias mitigation In this paper, we offer an augmentation to the usual release procedure for synthetic data by leveraging true and estimated importance weights. Most related to our work are the contributions from Elkan (2010) and Ji and Elkan (2013) who train a regularised logistic regression model and assign weights based on the Laplace-noise-contaminated coefficients of the logistic regression. In follow up work, Ji et al. (2014) propose to modify the update step of the Newton-Raphson optimisation algorithm used in fitting the logistic regression classifier to achieve DP. However, neither of these generalise well to more complex and high dimensional settings because of the linearity of the classifier. Further, the authors assume the existence of a *public dataset* while we consider the

case where we first generate DP *synthetic data* and then weight them a posteriori, providing a generic and universally applicable approach. The benefit of learning a generative model over using public data include on the one hand that there is no requirement for the existence of a public data set, and on the other hand the possibility to generate new data points. This distinction necessitates additional analysis as the privacy budget splits between the budget spent on fitting the SDGP and the budget for estimating the IW approach. Furthermore, we show that the approach from Ji and Elkan (2013) leads to statistically biased estimation and formulate an unbiased extension with improved properties.

3 DIFFERENTIAL PRIVACY AND IMPORTANCE WEIGHTING

From a decision theoretic perspective, the goal of statistics is estimating expectations of functions $h : X \mapsto \mathbb{R}$, e.g. loss or utility functions, w.r.t the distribution of future uncertainties $x \sim p_D$. Given data from $\{x'_1, \ldots, x'_{N_D}\} =: x'_{1:N_D} \overset{\text{i.i.d.}}{\sim} p_D$ the data analyst can estimate these expectations consistently via the strong law of large numbers as $\mathbb{E}_{x\sim p_D}(h(x)) \approx \frac{1}{N_D}\sum_{i=1}^{N_D} h(x'_i)$. However, under DP constraints the data analyst is no longer presented with a sample from the true DGP $x'_{1:N_D} \overset{\text{i.i.d.}}{\sim} p_D$ but with a synthetic data sample $x_{1:N_G}$ from the SDGP p_G. Applying the naive estimator in this scenario biases the downstream tasks as $\frac{1}{N_G}\sum_{i=1}^{N_G} h(x_i) \to \mathbb{E}_{x\sim p_G}(h(x))$ almost surely.

This bias can be mitigated using a standard Monte Carlo method known as importance weighting (IW). Suppose we had access to the weights $w(x) := \frac{p_D(x)}{p_G(x)}$. If $p_G(\cdot) > 0$ whenever $h(\cdot)p_D(\cdot) > 0$, then IW relies on

$$\mathbb{E}_{x\sim p_D}[h(x)] = \mathbb{E}_{x\sim p_G}\left[w(x)h(x)\right]. \tag{2}$$

So we have almost surely for $x_{1:N_G} \overset{\text{i.i.d.}}{\sim} p_G$ the convergence

$$\mathbb{E}_{x\sim p_D}[h(x)] = \mathbb{E}_{x\sim p_G}\left[w(x)h(x)\right].$$

So we have almost surely for $x_{1:N_G} \overset{\text{i.i.d.}}{\sim} p_G$ the convergence

$$I_N(h|w) := \frac{1}{N_G}\sum_{i=1}^{N_G} w(x_i)h(x_i) \overset{N_G\to\infty}{\longrightarrow} \mathbb{E}_{x\sim p_D}[h(x)].$$

3.1 IMPORTANCE WEIGHTED EMPIRICAL RISK MINIMISATION

A downstream task of particular interest is the use of $x'_{1:N_D} \sim p_D$ to learn a predictive model, $f(\cdot) \in \mathcal{F}$, for the data generating distribution p_D based on empirical risk minimisation. Given a loss function $h : \mathcal{F} \times X \mapsto \mathbb{R}$ comparing models $f(\cdot) \in \mathcal{F}$ with observations $x \in X$ and data $x'_{1:N_D} \sim p_D$, the principle of empirical risk minimisation (Vapnik, 1991) states that the optimal $\widehat{f}$ is given by the minimisation of

$$\frac{1}{N_D}\sum_{i=1}^{N_D} h(f(\cdot), x'_i) \approx \mathbb{E}_{x\sim p_D}\left[h(f(\cdot), x)\right]$$

over f. Maximum likelihood estimation (MLE) is a special case of the above with $h(f(\cdot), x_i) = -\log f(x_i|\theta)$ for a class of densities f parameterised by θ. Given synthetic data $x_{1:N_G} \sim p_G$, Equation (2) can be used to debias the learning of f.

Remark 1 (Supplement B.5)**.** *Minimisation of the importance weight adjusted log-likelihood, $-w(x_i)\log f(x_i|\theta)$, can be viewed as an M-estimator (e.g. Van der Vaart, 2000) with clear relations to the standard MLE.*

Bayesian updating. Wilde et al. (2020) showed that naively conducting Bayesian updating using DP synthetic data without any adjustment could have negative consequences for inference. To show the versatility of our approach and to address the issues they pointed out, we demonstrate how IW can help mitigate this. The posterior distribution for parameter θ given $\tilde{x}' := x'_{1:N_D} \sim p_D$ is

$$\pi(\theta|\tilde{x}') \propto \pi(\theta)\prod_{i=1}^{N_D} f(x'_i|\theta) = \pi(\theta)\exp\left(\sum_{i=1}^{N_D}\log f(x'_i|\theta)\right)$$

where $\pi(\theta)$ denotes the prior distribution for θ. This posterior is known to learn about model parameter $\theta^{\text{KLD}}_{p_D} := \arg\min_\theta \text{KLD}\left(p_D||f(\cdot|\theta)\right)$ (Berk, 1966; Bissiri et al., 2016) where KLD denotes the Kullback-Leibler divergence.

Given only synthetic data $\tilde{x} := x_{1:N_G}$ from the 'proposal distribution' p_G, we can use the importance weights defined in Equation (2) to construct the (generalised) posterior distribution

$$\pi_{IW}(\theta|\tilde{x}) \propto \pi(\theta)\exp\left(\sum_{i=1}^{N_G} w(x_i)\log f(x_i|\theta)\right). \tag{3}$$

In fact, Equation (3) corresponds to a generalised Bayesian posterior (Bissiri et al., 2016) with $\ell_{IW}(x_i;\theta) := -w(x_i)\log f(x_i|\theta)$, providing a coherent updating of beliefs about parameter $\theta^{\text{KLD}}_{p_D}$ using only data from the SDGP.

Theorem 1 (Supplement B.6)**.** *The importance weighted Bayesian posterior $\pi_{IW}(\theta|x_{1:N_G})$, defined in Equation (3) for $x_{1:N_G} \overset{\text{i.i.d.}}{\sim} p_G$, admits the same limiting Gaussian distribution as the Bayesian posterior $\pi(\theta|x'_{1:N_D})$ where $x'_{1:N_D} \overset{\text{i.i.d.}}{\sim} p_D$, under regularity conditions as in (Chernozhukov and Hong, 2003; Lyddon et al., 2018).*

It is necessary here to acknowledge the existence of methods to directly conduct privatised Bayesian updating (e.g.

Dimitrakakis et al., 2014; Foulds et al., 2016; Wang et al., 2015) or M-estimation (Avella-Medina, 2021). We refer the reader Section 1 for why the attention of this paper focuses on downstream tasks for private synthetic data. We consider the application of DP IW to Bayesian updating as a natural example of such a task.

3.2 ESTIMATING THE IMPORTANCE WEIGHTS

The previous section shows that IW can be used to recalibrate inference for synthetic data. Unfortunately, both the DGP p_D and SDGP p_G densities are typically unknown, e.g. due to the intractability of GAN generation, and thus the 'perfect' weight $w(x)$ cannot be calculated. Instead, we must rely on estimates of these weights, $\widehat{w}(x)$. In this section, we show that the existing approach to DP importance weight estimation is biased, and how the data curator can correct it.

Using the same reasoning as in Section 2.1, we argue that any calibrated classification method that learns to distinguish between data from the DGP, labelled thenceforth with $y = 1$, and from the SDGP, labelled with $y = 0$, can be used to estimate the likelihood ratio (Sugiyama et al., 2012). Using Equation (1), we compute

$$\widehat{w}(x) = \frac{\widehat{p}(y = 1|x)}{\widehat{p}(y = 0|x)} \frac{N_D}{N_G}$$

where $\widehat{p}$ are the probabilities estimated by such a classification algorithm. To improve numerical stability, we can also express the log weights as

$$\log \widehat{w}(x) = \sigma^{-1}(\widehat{p}(y = 1|x)) + \log \frac{N_D}{N_G},$$

where $\sigma(x) := (1 + \exp(-x))^{-1}$ is the logistic function and $\sigma^{-1}(\widehat{p}(y = 1|x))$ are the logits of the classification method. We will now discuss two such classifiers: logistic regression and neural networks.

3.3 PRIVATISING LOGISTIC REGRESSION

DP guarantees for a classification algorithm g can be achieved by adding noise to the training procedure. The scale of this noise is determined by how much the algorithm differs when one observation of the dataset changes. In more formal terms, the sensitivity of g w.r.t a norm $|\cdot|$ is defined by the smallest number $S(g)$ such that for any two neighbouring datasets $\mathcal{D}$ and $\mathcal{D}'$ it holds that

$$|g(\mathcal{D}) - g(\mathcal{D}')| \leq S(g).$$

Dwork et al. (2006) show that to ensure the differential privacy of g, it suffices to add Laplacian noise with standard deviation $S(g)/\epsilon$ to g.

Possibly the simplest classifier g one could use to estimate the importance weights is logistic regression with L_2 regularisation. It turns out this also has a convenient form for its sensitivity. If the data is scaled to a range from 0 to 1 such that $X \subset [0, 1]^d$, Chaudhuri et al. (2011) show that the L_2 sensitivity of the optimal coefficient vector estimated by $\widehat{\beta}$ in a regularised logistic regression with model

$$\widehat{p}(y = 1|x_i) = \sigma(\widehat{\beta}^T x_i) = \left(1 + e^{-\widehat{\beta}^T x_i}\right)^{-1}$$

is $S(\widehat{\beta}) = 2\sqrt{d}/(N_D\lambda)$ where λ is the coefficient of the L_2 regularisation term added to the loss during training. For completeness, when the logistic regression contains an intercept parameter, we let x_i denote the concatenation of the feature vector and the constant 1.

Ji and Elkan (2013) propose to compute DP importance weights by training such an L_2 regularised logistic classifier on the private and the synthetic data, and perturb the coefficient vector $\widehat{\beta}$ with Laplacian noise. For a d dimensional noise vector ζ with $\zeta_j \overset{i.i.d.}{\sim} \text{Laplace}(0, \rho)$ with $\rho = 2\sqrt{d}/(N_D\lambda\epsilon)$ for $j \in \{1, \ldots, d\}$, the private regression coefficient is then $\overline{\beta} = \widehat{\beta} + \zeta$, akin to adding heteroscedastic noise to the private estimates of the log weights

$$\log \overline{w}(x_i) = \overline{\beta}^T x_i = \widehat{\beta}^T x_i + \zeta x_i. \tag{4}$$

The resulting privatised importance weights can be shown to lead to statistically biased estimation.

Proposition 1 (Supplement B.1)**.** *Let $\overline{w}$ denote the importance weights computed by noise perturbing regression coefficients as in Equation* (4) *(Ji and Elkan, 2013, Algorithm 1). The IS estimator $I_N(h|\overline{w})$ is biased.*

Introducing bias on downstream estimators of sensitive information is undesirable as it can lead to an increased expected loss. To address this issue, we propose a fast and effective way for the data curator to debias the weights after computation, without requirement for an additional privacy budget.

Proposition 2 (Supplement B.2)**.** *Let $\overline{w}$ denote the importance weights computed by noise perturbing the regression coefficients as in Equation* (4) *(Ji and Elkan, 2013, Algorithm 1) where ζ can be sampled from any noise distribution that ensures (ϵ, δ)-differential privacy of $\overline{\beta}$. Define*

$$b(x_i) := 1/\mathbb{E}_{p_\zeta}[\exp\left(\zeta^T x_i\right)],$$

and adjusted importance weight

$$\overline{w}^*(x_i) = \overline{w}(x_i)b(x_i) = \widehat{w}(x_i) \exp\left(\zeta^T x_i\right) b(x_i). \tag{5}$$

The importance sampling estimator $I_N(h|\overline{w}^*)$ is unbiased and (ϵ, δ)-DP for $\mathbb{E}_{p_\zeta}[\exp\left(\zeta^T x_i\right)] > 0$.

In Supplement B.2.4, we further show that our approach does not only decrease the bias, but also the variance of the importance weighted estimators.

For the case of component-wise independent Laplace perturbations $\zeta_j \overset{i.i.d.}{\sim} \text{Laplace}(0, \rho)$, we show that the bias correction term can be computed as

$$b(x_i) = \prod_{j=1}^{d} \left(1 - \rho^2 x_{ij}^2\right), \text{ provided } |x_{ij}| < 1/\rho \quad \forall j.$$

In practice, e.g. as we observe empirically in Section 4, the optimal choice of the regularisation term λ is sufficiently large such that $\rho < 1$. Since the data is scaled to a range of 0 to 1 (Chaudhuri et al., 2011), this bias correction method is not limited by the restriction $|x_{ij}| < 1/\rho, \forall j$. If the data curator still encounters a case where this condition is not fulfilled, they can choose to perturb the weights with Gaussian noise instead, in which case the bias correction term always exists (see Supplement B.2.2). Laplacian perturbations are however preferred as the required noise scale can be expressed analytically without additional optimisation (Balle and Wang, 2018), and as they give stricter privacy guarantees with $\delta = 0$.

Alternatively, unbiased importance weighted estimates can be computed directly by noising the weights instead of the coefficients of the logistic regression. While this procedure removes the bias of the estimates and can also be shown to be consistent, it increases the variance to a greater extent than noising the coefficients does, and is thus only sustainable when small amounts of data are released. Please refer to Supplement A.1 for more details.

3.4 PRIVATISING NEURAL NETWORKS

If logistic regression fails to give accurate density ratio estimates, for example because of biases introduced by the classifier's linearity assumptions, a more complex discriminator in the form of a neural network can be trained. We can train DP classification neural networks for the aim of likelihood ratio estimation with stochastic gradient decent (SGD) by clipping the gradients and adding calibrated Gaussian noise at each step of the SGD, see e.g. Abadi et al. (2016). The noised gradients are then added up in a *lot* before the descent step where lots resemble mini-batches.

These optimisation algorithms are commonly formulated for the case when the complete dataset is private. However, in our setting, N_D observations are private and N_G observations are non-private. Thus, we can define a relaxed version of DP SGD. Algorithm 1 provides an overview of our proposed method. We highlight the modifications to Algorithm 1 from Abadi et al. (2016) in blue.

Proposition 3. *Each step in the SGD outlined in Algorithm 1 is (ϵ, δ)-differentially private w.r.t the lot and $(\mathcal{O}(q\epsilon), \delta)$ differentially private w.r.t the full dataset where $q = \frac{L}{N_D + N_G}$ and $\sigma = \sqrt{2 \log\left(\frac{1.25}{\delta}\right)}/\epsilon$.*

Algorithm 1: Relaxed DP SGD

Input: Examples $x_{1:N_D}, y_{1:N_D}$ from the DGP and $x_{N_D+1:N_D+N_G}, y_{N_D+1:N_D+N_G}$ from the SDGP, loss function $\mathcal{L}(\theta) = \frac{1}{N_G+N_D} \sum_i \mathcal{L}(\theta, x_i, y_i)$. Parameters: learning rate η_t, noise scale σ, expected lot size L, gradient norm bound C.

Initialise θ_0 randomly

for $t \in [T]$ **do**

  Construct a random subset $L_t \subset \{1, \ldots, N_D + N_G\}$ by including each index independently at random with probability $\frac{L}{N_D+N_G}$

  Compute gradient

  For each $i \in L_t$, compute $g_t(x_i, y_i) \leftarrow \Delta_{\theta_t} \mathcal{L}(\theta_t, x_i, y_i)$

  Clip gradient

  $\bar{g}_t(x_i, y_i) \leftarrow g_t(x_i, y_i) / \max(1, \frac{||g_t(x_i,y_i)||_2}{C})$

  Add noise

  $\tilde{g}_t \leftarrow \frac{1}{L} \sum_{i \in L_t} (\bar{g}_t(x_i, y_i) + N(0, \sigma^2 C^2 \mathbf{I}) \mathbb{1}_{(y_i=1)})$, where $\mathbb{1}_{(y_i=1)}$ is 1 if $y_i = 1$ and 0 otherwise

  Descent

  $\theta_{t+1} \leftarrow \theta_t + \eta_t \tilde{g}_t$

Output: θ_T and the overall privacy cost (ϵ, δ) using the moment's accountant of Abadi et al. (2016) with sampling probability $q = \frac{L}{N_D+N_G}$.

The differential privacy w.r.t a lot follows directly from the observation that the gradients of the synthetic data are already private. Further, the labels of the synthetic data are public knowledge. Lastly, the differential privacy w.r.t the dataset follows from the amplification theorem (Kasiviswanathan et al., 2011), the fact that sampling one particular private observation within a lot of size L is $q = \frac{L}{N_D+N_G}$, and the reasoning behind the moment accountant of Abadi et al. (2016). We still clip the gradients of the public dataset as their influence will otherwise be overproportional under strong maximum norm assumptions.

3.5 GAN DISCRIMINATOR WEIGHTS

The downside of the aforementioned likelihood ratio estimators (Equation (4), Equation (5), and Algorithm 1) is that their training requires an additional privacy budget which has to be added to the privacy budget used to learn the SDGP. If we however use a GAN such as DPGAN or PATE-GAN for private synthetic data generation, we can use the GAN's discriminator for the computation of the importance weights. According to the post processing theorem, these importance weights can be released without requiring an additional pri-

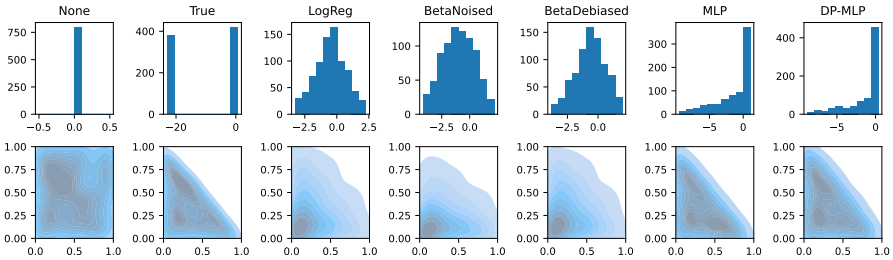

Figure 1: Kernel density plots of 100 observations sampled from a two dimensional uniform square distribution as SDGP (bottom left) and a uniform triangle distribution as DGP (second figure in second row). The first row depicts histograms of the computed weights starting with the true importance weights (True). The DP weights were privatised with $\epsilon = 1$, and the regularisation was chosen as $\lambda = 0.1$. The second row illustrates the importance weighted synthetic observations. We observe that while BetaDebiased corrects the weights of the logistic regression, the complex nature of the MLPs allows a better modelling of the DGP even in this simple setting.

vacy budget. In contrast to the weights computed from DP classification networks, the weights from this approach are thus more robust and require no additional hyperparameter tuning (confer to Section 4).

4 EXPERIMENTS

We demonstrate the benefits of using debiased IW for DP data release with a large-scale experimental study comparing three different SDGPs (DPGAN, DPCGAN, PrivBayes) on six real-world data sets (Iris, TGFB, Boston, Breast, Banknote, MNIST) for two different privacy budgets, $\epsilon \in \{1, 6\}$. We stress that debiasing comes with little overhead to the actual computations. As we see in Supplement C.2, the computations of the logistic regression and neural network importance weight estimates take less than one and a half minutes to train, even on MNIST. These weight estimators can be applied to any kind of synthetic data generation model, while the importance weights of the GAN discriminator can be computed in a single line of Python code and do not require any additional concerns regarding the privacy budget. Please see `https://github.com/sghalebikesabi/importance-weighted-differential-privacy` for the implementation.

Computation of importance weights After fitting the SDGP on the scaled true data, we weight each synthetic observation with importance weights. Based on the train and the synthetic data, we apply one of the following IW approaches: weights computed from a non-private logistic regression (LogReg), its DP alternative introduced by Ji and Elkan (2013) (BetaNoised), or our debiased proposal (BetaDebiased), and likelihood ratios estimated by a non-private multi-layer perceptron (MLP), or a DP-MLP trained using Algorithm 1. We also compare to the naive estimator using uniform weights without IW (called 'None').

Please refer to Supplement C.1 for more details on the implementation and the hyperparameters used in our experiments. In Supplement C.8, we provide a comparison to the experimental results reported by related papers. Because of the large scale of our experimental study, we present only the most important results in this section, and give a complete overview in Supplement C. The code and data for all experiments can be found online.

4.1 TOY EXAMPLE

We start our analysis with a simple example to illustrate the benefits of the different weighting schemes. We assume that the synthetic data is sampled from a two-dimensional uniform distribution from 0 to 1 whereas the true data follows a uniform distribution on the lower triangle given by $x_1 + x_2 < 1$ for $x_1, x_2 \in [0, 1]$. This illustrative toy example was chosen for a fairer comparison of the logistic regression and the neural network based approaches. As we see in Figure 1, the weighted kernel density estimate (KDE) of BetaDebiased is closer to the LogReg weighted KDE, and also the true KDE compared to the BetaNoised KDE.

4.2 UCI DATA SETS

Datasets and preprocessing We performed additional experiments on four UCI datasets of different characteristics as decribed in Supplement C.1: Iris, Banknote, Boston, and Breast. Similarly to Chaudhuri et al. (2011); Ji and Elkan (2013), we scale all data to a feature range from 0 to 1. We use a train-test split of 80%. In all experiments we fix δ to $N_D^{-1} - 10^{-6}$, and choose $\epsilon \in \{1, 6\}$. We refer to Supplement C.7 for a complete overview of the results.

Synthetic data generators We used DPCGAN (Torkzadehmahani et al., 2019), DPGAN (Xie et al., 2018), and their corresponding non-DP analogues (CGAN and CGAN) to generate DP synthetic data of the same size as the training

	IW	Breast			Banknote		
		DPGAN	DPCGAN	PrivBayes	DPGAN	DPCGAN	PrivBayes
WST ↓	None	$2.3665_{\pm 0.0982}$	$1.5853_{\pm 0.1333}$	$2.1117_{\pm 0.1740}$	$0.4746_{\pm 0.0214}$	$0.7442_{\pm 0.0333}$	$0.3237_{\pm 0.0162}$
	BetaNoised	$1.4337_{\pm 0.1114}$	$2.2232_{\pm 0.2325}$	$1.2322_{\pm 0.0823}$	$0.2509_{\pm 0.0436}$	$0.4355_{\pm 0.0456}$	$0.2318_{\pm 0.0035}$
	BetaDebiased	$1.8922_{\pm 0.1237}$	$1.9913_{\pm 0.3507}$	$\mathbf{1.1825_{\pm 0.0933}}$	$0.4015_{\pm 0.0766}$	$0.4618_{\pm 0.0832}$	$0.2369_{\pm 0.0061}$
	DP-MLP	$1.4570_{\pm 0.1492}$	$1.0315_{\pm 0.1415}$	$1.2190_{\pm 0.0795}$	$\mathbf{0.2035_{\pm 0.0427}}$	$0.4298_{\pm 0.0433}$	$\mathbf{0.0456_{\pm 0.0061}}$
	Discriminator	$\mathbf{1.0007_{\pm 0.0004}}$	$\mathbf{1.0001_{\pm 0.0001}}$	-	$0.3382_{\pm 0.0399}$	$\mathbf{0.1087_{\pm 0.0415}}$	-
	LogReg	$1.6451_{\pm 0.1168}$	$2.2953_{\pm 0.2121}$	$1.4663_{\pm 0.1152}$	$0.2508_{\pm 0.0432}$	$0.4348_{\pm 0.0460}$	$0.2348_{\pm 0.0034}$
	MLP	$1.6129_{\pm 0.1404}$	$1.0709_{\pm 0.1579}$	$1.4141_{\pm 0.1216}$	$0.0913_{\pm 0.0259}$	$0.3860_{\pm 0.0452}$	$0.0021_{\pm 0.0004}$
β MSE ↓	None	$2.0643_{\pm 0.2012}$	$4.9828_{\pm 1.5701}$	$2.3904_{\pm 0.1050}$	$11.0215_{\pm 1.8377}$	$19.3243_{\pm 3.7708}$	$8.1724_{\pm 0.3987}$
	BetaNoised	$2.7532_{\pm 0.2650}$	$2.5025_{\pm 0.3763}$	$2.1144_{\pm 0.2400}$	$8.4298_{\pm 1.0383}$	$15.2862_{\pm 4.0365}$	$5.7001_{\pm 0.1885}$
	BetaDebiased	$2.8337_{\pm 0.3842}$	$\mathbf{2.2324_{\pm 1.0446}}$	$\mathbf{1.8266_{\pm 0.2392}}$	$\mathbf{8.3508_{\pm 2.3127}}$	$12.9909_{\pm 5.9024}$	$6.6862_{\pm 0.1458}$
	DP-MLP	$2.3965_{\pm 0.2083}$	$3.8865_{\pm 0.6043}$	$2.3130_{\pm 0.2195}$	$17.1597_{\pm 2.5448}$	$16.4618_{\pm 4.1011}$	$\mathbf{3.5519_{\pm 0.2895}}$
	Discriminator	$\mathbf{1.4591_{\pm 0.1837}}$	$4.0612_{\pm 0.9523}$	-	$12.5471_{\pm 2.3124}$	$\mathbf{10.9282_{\pm 5.4283}}$	-
	LogReg	$2.6934_{\pm 0.2667}$	$2.2156_{\pm 0.3366}$	$1.5333_{\pm 0.2138}$	$8.4760_{\pm 1.0406}$	$15.2964_{\pm 4.0396}$	$5.6751_{\pm 0.1785}$
	MLP	$2.3999_{\pm 0.2040}$	$3.8343_{\pm 0.7032}$	$1.6581_{\pm 0.2020}$	$17.9390_{\pm 2.4926}$	$15.5211_{\pm 4.2147}$	$2.6286_{\pm 0.3761}$
MLP ROC-AUC ↑	None	$0.6374_{\pm 0.0421}$	$0.6791_{\pm 0.0966}$	$0.8366_{\pm 0.0579}$	$0.8546_{\pm 0.0213}$	$0.6863_{\pm 0.0436}$	$0.7630_{\pm 0.0495}$
	BetaNoised	$0.6110_{\pm 0.0477}$	$0.6546_{\pm 0.0727}$	$0.7076_{\pm 0.0983}$	$0.8495_{\pm 0.0274}$	$0.6063_{\pm 0.0510}$	$0.8943_{\pm 0.0173}$
	BetaDebiased	$0.6820_{\pm 0.0510}$	$0.7173_{\pm 0.0842}$	$\mathbf{0.8557_{\pm 0.0765}}$	$\mathbf{0.8729_{\pm 0.0310}}$	$0.5868_{\pm 0.1005}$	$0.7632_{\pm 0.0517}$
	DP-MLP	$\mathbf{0.7942_{\pm 0.0404}}$	$0.5686_{\pm 0.0823}$	$0.7353_{\pm 0.0887}$	$0.7697_{\pm 0.0419}$	$0.5657_{\pm 0.0570}$	$\mathbf{0.8953_{\pm 0.0299}}$
	Discriminator	$0.6992_{\pm 0.0839}$	$\mathbf{0.7290_{\pm 0.0720}}$	-	$0.8695_{\pm 0.0167}$	$\mathbf{0.7114_{\pm 0.0424}}$	-
	LogReg	$0.6631_{\pm 0.0469}$	$0.6484_{\pm 0.1081}$	$0.7618_{\pm 0.1019}$	$0.8172_{\pm 0.0327}$	$0.6034_{\pm 0.0534}$	$0.9102_{\pm 0.0129}$
	MLP	$0.7730_{\pm 0.0412}$	$0.7358_{\pm 0.1017}$	$0.7573_{\pm 0.0738}$	$0.8291_{\pm 0.0333}$	$0.5974_{\pm 0.0627}$	$0.8594_{\pm 0.0231}$

Table 1: Mean and standard error over 10 runs for ($\epsilon = 1$, $\delta = N_D^{-1} - e^{-6}$) on the Breast and Banknote data. Best score out of the private methods is marked in bold.

data set. Additionally we also consider PrivBayes (Zhang et al., 2017), a DP Bayesian Network, as a potential SDGP.

Hyperparameter tuning Hyperparameter tuning is essentially non-private, and has to be accounted for in the privacy budget. Since hyperparemeter tuning in a DP setting is an unresolved problem (Liu and Talwar, 2019; Rosenblatt et al., 2020; Papernot and Steinke, 2021), we follow Jordon et al. (2019) and tune the hyperparameters of the underlying baselines on private validation data sets. However, we propose default parameters for our methods. This leads to an over-optimistic presentation of the baseline performance, and a conservative presentation of our extensions.

Evaluation metrics In order to show that IW decreases statistical bias, we train a linear prediction model on the synthetic data and approximate its bias. Since the true DGP is not known, we train the same linear predictor on the test data and report the mean squared error (MSE) between the test parameters and the parameters estimated on the SDGP, as β MSE. We further analyse the divergence of the weighted SDGP and the DGP in a similar way by computing the Wassertstein (WST) distance w.r.t the test data. As one exemplary supervised downstream task, we consider a linear downstream classifier or regressor trained on the synthetic data. This downstream predictor is then assessed by the error measured in the parameter vector compared to the parameters learnt using the test set ($beta$ MSE). Finally, we train a one-hidden-layer MLP on the training data, and report the test prediction error as MLP ROC-AUC for classification tasks, and MLP MSE for regression tasks.

Choice of budget split We only present results for $\epsilon = 1$ in this section, and refer the reader to Supplement C.7 for further results with $\epsilon = 6$. If the weight computation procedure requires a separate privacy budget (e.g. if the weights are computed by a separate MLP or logistic regression), we spend 10% of the ϵ-budget on fitting the SDGP and 30% of the δ-budget on the weight computation; the complete budget can be spent on fitting the SDGP if no weights, or the weights of the discriminator are used. In Supplement C.3, we evaluate a range of different privacy splits on the Breast and Boston data.

Results In Tables 1 and 2, we see that the performance of the models mostly improved when weighted with any type of estimated weights. Although the best inference for each data set is nearly always achieved after importance weighting, we notice that there are some rare cases where no importance weighting performs (insignificantly) better. For instance, we observe that the SDGP obtained with PrivBayes seems to be close to the true DGP of the Boston Housing data, and that importance weighting is no longer helpful. In settings where the SDGP and the DGP are really close, it is possible that the effects of additional variance induced by estimating and privatising the importance weights (where appropriate) cancels out the reduction in bias. This effect might be mitigated with hyperparameter tuning. Further, we note that debiasing the logistic regression weights mainly results in better performance. Even though we experience a slight

	IW	DPGAN	PrivBayes
WST ↓	None	$2.2013_{\pm 0.0945}$	$1.3938_{\pm 0.0231}$
	BetaNoised	$2.0922_{\pm 0.0419}$	$1.3009_{\pm 0.0338}$
	BetaDebiased	$2.0930_{\pm 0.0393}$	$1.2705_{\pm 0.0290}$
	DP-MLP	$2.0542_{\pm 0.0184}$	$\mathbf{1.0265_{\pm 0.0035}}$
	Discriminator	$\mathbf{2.0145_{\pm 0.0141}}$	-
	LogReg	$2.2051_{\pm 0.0819}$	$1.4078_{\pm 0.0492}$
	MLP	$2.0350_{\pm 0.0158}$	$1.0072_{\pm 0.0009}$
β MSE ↓	None	$0.1867_{\pm 0.0434}$	$\mathbf{0.0011_{\pm 0.0002}}$
	BetaNoised	$0.1761_{\pm 0.0948}$	$0.0088_{\pm 0.0028}$
	BetaDebiased	$\mathbf{0.0667_{\pm 0.0188}}$	$0.0077_{\pm 0.0022}$
	DP-MLP	$0.1530_{\pm 0.0812}$	$0.0048_{\pm 0.0024}$
	Discriminator	$0.1567_{\pm 0.1825}$	-
	LogReg	$0.0749_{\pm 0.0279}$	$0.0037_{\pm 0.0016}$
	MLP	$0.1476_{\pm 0.0804}$	$0.0008_{\pm 0.0002}$
MLP MSE ↓	None	$1.8851_{\pm 0.5262}$	$0.1973_{\pm 0.0108}$
	BetaNoised	$1.0057_{\pm 0.1973}$	$0.2200_{\pm 0.0154}$
	BetaDebiased	$\mathbf{0.9024_{\pm 0.1244}}$	$0.2139_{\pm 0.0122}$
	DP-MLP	$0.9462_{\pm 0.1702}$	$\mathbf{0.1877_{\pm 0.0174}}$
	Discriminator	$1.6256_{\pm 0.2394}$	-
	LogReg	$1.0606_{\pm 0.2648}$	$0.2515_{\pm 0.0305}$
	MLP	$1.0979_{\pm 0.2225}$	$0.1697_{\pm 0.0079}$

Table 2: Mean and standard error over 10 runs for ($\epsilon = 1$, $\delta = N_D^{-1} - e^{-6}$) on the Boston Housing data. Best score out of the private methods is marked in bold.

IW	β MSE ↓	MLP ROC-AUC ↑
None	$0.6605_{\pm 0.0384}$	$0.8502_{\pm 0.0386}$
BetaNoised	$0.6247_{\pm 0.0184}$	$0.8766_{\pm 0.0086}$
BetaDebiased	$0.6240_{\pm 0.0179}$	$\mathbf{0.8783_{\pm 0.0093}}$
DP-MLP	$\mathbf{0.5813_{\pm 0.0246}}$	$0.8683_{\pm 0.0055}$
Discriminator	$0.6242_{\pm 0.0140}$	$0.8631_{\pm 0.0310}$
LogReg	$0.6234_{\pm 0.0183}$	$0.8770_{\pm 0.0092}$
MLP	$0.5707_{\pm 0.0207}$	$0.8737_{\pm 0.0058}$

Table 3: Mean and standard error over 10 runs with standard errors for ($\epsilon = 9.64, \delta = 60,000^{-1} - e^{-6}$) on MNIST.

drop in performance from BetaNoised to BetaDebiased in some rare cases, this can be explained by randomness in the data set as we show in Supplement Table 3 that the weights estimated by BetaDebiased are significantly closer to the true LogReg weights than the importance weights given by BetaNoised. If a GAN is used as SDGP, and the data curator is hesitant to release additional importance weights, the discriminator weights nearly always lead to an improvement in results without requiring additional computations. To further illustrate the practical meaning of debiasing, we have included an exemplary case study in Supplement C.6.

4.3 BAYESIAN UPDATING WITH IW

We investigate the effectiveness of IW in a Bayesian learning setting as per Equation 3. We evaluated and compared the performance of these weighted posteriors alongside the standard non-weighted posterior by applying them to learning the parameters of models for various regression tasks. Figure 2 shows the ROC-AUC scores associated with the Bayesian predictive distribution arising from integration over the posterior of a Bayesian logistic regression model fit on synthesised versions of the Banknote dataset. We observe that the ROC-AUC under PrivBayes' synthetic data is significantly improved upon across all IW methods, with similar gains made to the median performance under CGAN's synthetic data. Additionally, most of the methods help in decreasing variability in the results, especially DP-MLP and MLP. See Supplement C.5 for a full specification of the experimental details and for further results from fitting Bayesian linear regression and multinomial logistic regression models on the TGFB and Iris datasets respectively.

4.4 MNIST

Additionally, we assessed how IW performs in a high-dimensional setting such as a classification task on the MNIST dataset. Since PrivBayes does not scale to large data sets, we only evaluate DPCGAN as possible SDGP. For this we follow the setup by Torkzadehmahani et al. (2019) for $\epsilon = 9.64$ and $\delta = 6000^{-1} - 10^{-6}$. We observe in Table 3 that all IW methods improve upon the state of the art.

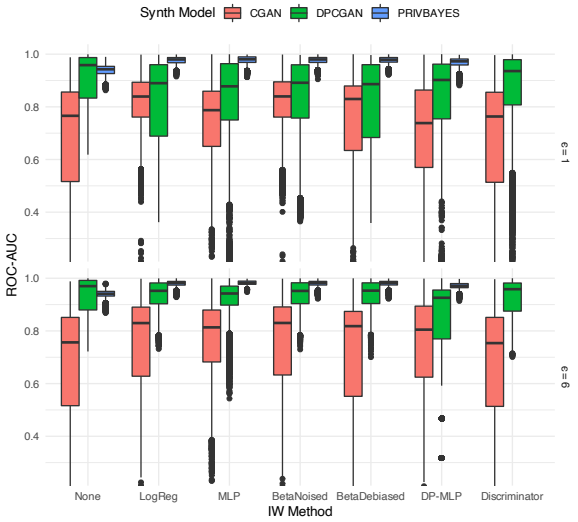

Figure 2: ROC-AUC score box plots calculated via chains of parameters sampled from a Bayesian logistic regression model fit on synthesised Banknote data across 10 seeds.

5 DISCUSSION

In this paper, we investigated importance weighting methods to correct for biases in downstream estimation tasks when using differentially private synthetic data. While classification algorithms can be used to estimate the required importance

weights, noise must be added in order to maintain privacy. We presented methods to debias inference based on privatised weights estimated by logistic regression, developed private estimation procedures allowing the complexity of neural networks to be leveraged for weight estimation, and proposed using inbuilt discriminator weights from GAN data generation to avoid increases to the privacy budget.

Following these developments, we advocate that future releases of DP synthetic data are augmented with privatised importance weights to allow researchers to conduct unbiased downstream model estimation. Future work will focus on improved hyperparameter tuning practises to choose the optimal IW approach for the task and dataset at hand. Further improving upon the likelihood ratio estimates in a non-private setting could simplify such a choice.

Acknowledgements

SG is a student of the EPSRC CDT in Modern Statistics and Statistical Machine Learning (EP/S023151/1) and receives funding from the Oxford Radcliffe Scholarship and Novartis. HW is supported by the Feuer International Scholarship in Artificial Intelligence. JJ was funded by the Ayudas Fundación BBVA a Equipos de Investigación Cientifica 2017 and Government of Spain's Plan Nacional PGC2018-101643-B-I00 grants whilst working on this project. SJV is supported by the University of Warwick, University of Warwick and German Resarch Centre for Aritifical Intelligence. CH is supported by The Alan Turing Institute, Health Data Research UK, the Medical Research Council UK, the EPSRC through the Bayes4Health programme Grant EP/R018561/1, and AI for Science and Government UK Research and Innovation (UKRI).

References

Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318, 2016.

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

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

Marco Avella-Medina. Privacy-preserving parametric inference: a case for robust statistics. *Journal of the American Statistical Association*, 116(534):969–983, 2021.

Samaneh Azadi, Catherine Olsson, Trevor Darrell, Ian Goodfellow, and Augustus Odena. Discriminator rejection sampling. *arXiv preprint arXiv:1810.06758*, 2018.

Borja Balle and Yu-Xiang Wang. Improving the gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In *International Conference on Machine Learning*, pages 394–403. PMLR, 2018.

Robert H Berk. Limiting behavior of posterior distributions when the model is incorrect. *The Annals of Mathematical Statistics*, pages 51–58, 1966.

Pier Bissiri, Chris Holmes, and Stephen Walker. A general framework for updating belief distributions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 2016.

Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(3), 2011.

Qingrong Chen, Chong Xiang, Minhui Xue, Bo Li, Nikita Borisov, Dali Kaarfar, and Haojin Zhu. Differentially private data generative models. *arXiv preprint arXiv:1812.02274*, 2018.

Victor Chernozhukov and Han Hong. An MCMC approach to classical estimation. *Journal of Econometrics*, 115(2): 293–346, 2003.

Christos Dimitrakakis, Blaine Nelson, Aikaterini Mitrokotsa, and Benjamin IP Rubinstein. Robust and private bayesian inference. In *International Conference on Algorithmic Learning Theory*, pages 291–305. Springer, 2014.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

Charles Elkan. Preserving privacy in data mining via importance weighting. In *International Workshop on Privacy and Security Issues in Data Mining and Machine Learning*, pages 15–21. Springer, 2010.

Liyue Fan. A survey of differentially private generative adversarial networks. In *The AAAI Workshop on Privacy-Preserving Artificial Intelligence*, 2020.

James Foulds, Joseph Geumlek, Max Welling, and Kamalika Chaudhuri. On the theory and practice of privacy-preserving bayesian data analysis. *arXiv preprint arXiv:1603.07294*, 2016.

Lorenzo Frigerio, Anderson Santana de Oliveira, Laurent Gomez, and Patrick Duverger. Differentially private generative adversarial networks for time series, continuous, and discrete open data. In *IFIP International Conference on ICT Systems Security and Privacy Protection*, pages 151–164. Springer, 2019.

Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014.

Aditya Grover, Jiaming Song, Ashish Kapoor, Kenneth Tran, Alekh Agarwal, Eric J Horvitz, and Stefano Ermon. Bias correction of learned generative models using likelihood-free importance weighting. In *Advances in Neural Information Processing Systems*, pages 11058–11070, 2019.

Moritz Hardt and Guy N Rothblum. A multiplicative weights mechanism for privacy-preserving data analysis. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 61–70. IEEE, 2010.

Stephanie Hyland, Cristóbal Esteban, and Gunnar Rätsch. Real-valued (medical) time series generation with recurrent conditional gans. *arXiv*, 2018.

Zhanglong Ji and Charles Elkan. Differential privacy based on importance weighting. *Machine Learning*, 93(1):163–183, 2013.

Zhanglong Ji, Xiaoqian Jiang, Shuang Wang, Li Xiong, and Lucila Ohno-Machado. Differentially private distributed logistic regression using private and public data. *BMC medical genomics*, 7(1):1–10, 2014.

James Jordon, Jinsung Yoon, and Mihaela van der Schaar. Pate-gan: Generating synthetic data with differential privacy guarantees. In *International Conference on Learning Representations*, 2019.

Shiva Prasad Kasiviswanathan, Homin K Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? *SIAM Journal on Computing*, 40(3): 793–826, 2011.

Moshe Lichman. UCI machine learning repository, 2013.

Jingcheng Liu and Kunal Talwar. Private selection from private candidates. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 298–309, 2019.

Terrance Liu, Giuseppe Vietri, Thomas Steinke, Jonathan Ullman, and Steven Wu. Leveraging public data for practical private query release. In *International Conference on Machine Learning*, pages 6968–6977. PMLR, 2021.

Simon P Lyddon, Chris Holmes, and Stephen Walker. General Bayesian updating and the loss-likelihood bootstrap. *Biometrika*, 2018.

Marcel Neunhoeffer, Zhiwei Steven Wu, and Cynthia Dwork. Private post-GAN boosting. *arXiv preprint arXiv:2007.11934*, 2020.

Nicolas Papernot and Thomas Steinke. Hyperparameter tuning with renyi differential privacy. *arXiv preprint arXiv:2110.03620*, 2021.

Lucas Rosenblatt, Xiaoyan Liu, Samira Pouyanfar, Eduardo de Leon, Anuj Desai, and Joshua Allen. Differentially Private Synthetic Data: Applied Evaluations and Enhancements. *arXiv*, Nov 2020. URL `https://arxiv.org/abs/2011.05537v1`.

Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. *Density Ratio Estimation in Machine Learning*. Cambridge University Press, 2012.

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

Ryan Turner, Jane Hung, Eric Frank, Yunus Saatchi, and Jason Yosinski. Metropolis–Hastings generative adversarial networks. In *International Conference on Machine Learning*, pages 6345–6353. PMLR, 2019.

Aad W Van der Vaart. *Asymptotic Statistics*, volume 3. Cambridge University Press, 2000.

Yu-Xiang Wang, Stephen Fienberg, and Alex Smola. Privacy for free: Posterior sampling and stochastic gradient monte carlo. In *International Conference on Machine Learning*, pages 2493–2502. PMLR, 2015.

Harrison Wilde, Jack Jewson, Sebastian Vollmer, and Chris Holmes. Foundations of Bayesian learning from synthetic data. *arXiv preprint arXiv:2011.08299*, 2020.

Liyang Xie, Kaixiang Lin, Shu Wang, Fei Wang, and Jiayu Zhou. Differentially private generative adversarial network. *arXiv preprint arXiv:1802.06739*, 2018.

Jun Zhang, Graham Cormode, Cecilia M Procopiuc, Divesh Srivastava, and Xiaokui Xiao. PrivBayes: Private data release via Bayesian networks. *ACM Transactions on Database Systems (TODS)*, 42(4):1–41, 2017.

Xinyang Zhang, Shouling Ji, and Ting Wang. Differentially private releasing via deep generative model. *arXiv preprint arXiv:1801.01594*, 2018.