Differentially Private Normalizing Flows for Privacy-Preserving Density Estimation

Chris Waites
Stanford University
Stanford, CA 94305
waites@stanford.edu

Abstract

Normalizing flow models have risen as a popular solution to the problem of density estimation, enabling the ability to perform both high-quality synthetic data generation as well as exact probability density evaluation. However, in contexts where individuals are directly associated with the training data at hand, releasing such a model raises natural privacy concerns. In this work, we propose the use of normalizing flow models that provide explicit differential privacy guarantees as a novel approach to the problem of privacy-preserving density estimation. We evaluate the efficacy of such an approach empirically using benchmark datasets and demonstrate that the proposed method substantially outperforms previous state-of-the-art approaches.

1 Introduction

The task of density estimation concerns the construction of an estimate, given observed data, of an unknown probability density function. Typically the construction of this estimate allows one to perform a variety of tasks of interest, including log likelihood evaluation as well as synthetic data generation. In contexts concerning sensitive data, the construction and subsequent release of such an estimate could very well leak potentially private information. For example, without explicitly asserting a rigorous privacy guarantee, nothing precludes the possibility of an individual's data appearing in the synthetic data generated by the model, disproportionate density being assigned to a point corresponding to them, or any other vulnerability due to arbitrary analysis of the learned model parameters. Since density estimation remains a task of interest to the modeling community, continued attention is required to address how such approaches respect participant privacy.

Differential privacy [15] has emerged as the predominant notion for privacy in the context of statistical data analysis. At a high level, differentially private analyses assert a bound on the extent to which their output distribution can change due to the inclusion or exclusion of any one individual from the analysis. Algorithms which adhere to this notion exhibit a number of desirable properties, including privacy guarantees which hold regardless of the auxiliary information an adversary may have and composition of privacy guarantees across multiple analyses. Hence differential privacy acts as a compelling gold standard in the design of privacy-preserving analyses.

Tools for density estimation are of longstanding interest due to their versatility. Their ability to address a wide range of tasks concerning a distribution is precisely why the existence of an accurate and privacy-preserving density estimation would be surprising. For example, the private construction of such a model implicitly yields a differentially private approach to anomaly detection—a task of substantial previous investigation [3, 33, 18]—through an immediate application of likelihood evaluation. In addition, given that density estimators often enable efficient sampling, such a model would yield a viable method for privacy-preserving synthetic data generation. This task in particular has been of longstanding interest to the privacy community [39] as it addresses many of the limitations

Preprint. Under review.

imposed by the query model [14] by allowing large numbers of arbitrary analyses. Privately generating a synthetic dataset only incurs a fixed privacy cost during the generation process; all subsequent queries on the synthetic data are automatically differentially private due to the privacy notion's post-processing guarantee, so the privacy cost does not scale with the number of analyses performed.

Normalizing flow models present themselves as a particularly attractive approach to the task of density estimation due to their proven empirical ability to approximate arbitrary, high-dimensional distributions. These models approach the task of density estimation via a transformation on a chosen base density by a sequence of invertible, non-linear transformations, enabling density querying on the resulting distribution via an application of the change-of-variables formula. Approaches to density estimation in this manner include: Non-linear Independent Components Estimation (NICE) [9], Real NVP [10], Glow [27], and Masked Autoregressive Flows (MAF) [36]. It was an open question whether normalizing flow models could be constructed in a differentially private manner to handle the task of privacy-preserving density estimation, combining the rigorous guarantees of differential privacy with the strong empirical performance exhibited by normalizing flows.

In this work we propose the use of normalizing flow models, trained in a differentially private manner, as a novel approach to the task of privacy-preserving density estimation. We outline an algorithm (DP-NF, Algorithm 1 in Section 3) that privately optimizes the model parameters via gradient descent according to DP-SGD [1], an application of Gaussian noise to clipped gradient updates which achieves differential privacy guarantees. Additionally, rather than perform composition under Rényi differential privacy [31] using the moments accountant (MA) [1], we achieve tighter privacy guarantees via composition under the recently introduced notion of Gaussian differential privacy [11]. We apply this optimization to the parameters of a Masked Autoregressive Flow [36], our primary architecture of consideration, and achieve empirical results (Section 4) which significantly outperform relevant previous approaches.

1.1 Related Work

Gaussian mixture models (GMMs) are known to be a particularly strong density estimation baseline [35] given that they are a *universal approximator of densities* - that is, they are able to approximate any density function arbitrarily well given a sufficient number of components [30]. They approach the task of density estimation through a weighted sum of Gaussian distributions, parameterized in full by their respective means, covariance matrices, and weights. The first differentially private algorithm for learning the parameters of a Gaussian mixture model comes from the work of [32] which makes use of their *sample-and-aggregate* framework to convert non-private algorithms into private algorithms, applied to the task of learning mixtures of Gaussians. However, their approach exhibits strong assumptions on the range of the parameter space and assumes a uniform mixture of spherical Gaussians in their investigation. Follow-up work of [25] proposes a modernized approach which improves upon the sample complexity of the aforementioned work and removes the strong a priori bounds on the parameters of the mixture components, although it makes the assumption that the components of the mixture are sufficiently well-separated.

There has also been work in learning the parameters of a Gaussian mixture model through differentially private variants of expectation maximization (EM). One notable instance of this is DPGMM [41], which achieves a privacy guarantee at each iteration of EM through the addition of calibrated Laplace noise to the estimated parameters following the maximization step. These individual privacy guarantees are then combined into an overall privacy guarantee via sequential composition, i.e., by taking their sum. The work of [37] follows a conceptually similar approach of applying either calibrated Laplace or Gaussian noise to the parameters of the model at the end of each EM iteration, but demonstrates significantly better privacy guarantees through composition via the moments accountant and zero-concentrated differential privacy (zCDP) [6]. Given that their work makes no significant assumptions about the task and provides an empirical evaluation of their methods, this is likely the closest in nature to our approach. As such, is included as a baseline in our experimental results.

In addition, we take note of more classical approaches to the task of privacy-preserving density estimation. One of the simplest yet most widely used methods for density estimation is through the use of histograms, and previous work [8] has investigated their private estimation. Unfortunately, such an approach scales poorly with the dimension and complexity of the distribution while asserting an unrealistic discretization of the space. Kernel density estimation is another closely related approach, often characterized as the smooth analog to the classical discrete histogram. The work of [22]

proposes a method for privately querying the density of such an estimator through the addition of calibrated Gaussian noise. As a non-parametric approach, it has the drawback that it requires storage of the entire dataset at test time to enable querying (proving impractical for large-scale datasets) while still degrading similarly with dimension.

We also include a brief overview of the extensive literature concerning density estimation via normalizing flows. One important subset are those characterized by *coupling layers*: transformations which partition the dimensions of its input and map them in an way which retains invertibility while maintaining a tractable Jacobian. This includes Non-linear Independent Components Estimation (NICE) [9], as well as its subsequent generalization Real NVP [10]. Another notable approach, Glow [27], makes use of such coupling layers while also proposing the use of an invertible weight matrix decomposition to generalize the notion of permutation layers. Alternatively, some make use of *autoregressive transformations*, which are transformations that utilize the chain rule of probability to represent a joint distribution as a product of its conditionals. Such models include Masked Autoregressive Flow (MAF) [36], a generalization of Real NVP optimized for density estimation, as well as its closely related Inverse Autoregressive Flow [28] optimized for variational inference, among others [34, 23].

2 Preliminaries

2.1 Normalizing Flows

Let $p(\cdot)$ be the probability density function characterizing an unobservable distribution of interest, and let $\boldsymbol{X} = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(n)}\}$ be n observed i.i.d. samples from this distribution. The task of density estimation is to find an approximation of $p(\cdot)$ via some model $p_{\boldsymbol{\theta}}(\cdot)$ given $\boldsymbol{X}$. In the context of normalizing flows, this model is characterized by a prior distribution $q(\cdot)$, chosen to exhibit a simple and tractable density (e.g. the spherical multivariate Gaussian distribution), and a sequence of K bijective functions $f_{\boldsymbol{\theta}} = f_1 \circ f_2 \circ \ldots f_K$, in full parameterized by $\boldsymbol{\theta}$. $f_{\boldsymbol{\theta}}$ in this case acts as a transformation between the prior distribution $q(\cdot)$ and the approximated distribution $p_{\boldsymbol{\theta}}(\cdot)$.

Given such a model, it can be used to efficiently sample $\boldsymbol{x} \sim p_{\boldsymbol{\theta}}$ by first sampling $\boldsymbol{z} \sim q$ and then transforming the sample as $\boldsymbol{x} = f_{\boldsymbol{\theta}}(\boldsymbol{z})$. If $p_{\boldsymbol{\theta}}$ is a good approximation of p, then this generative process gives an efficient (approximate) oracle for sampling from the unknown distribution.

Since $f_{\boldsymbol{\theta}}$ is invertible, one can also perform exact likelihood evaluation on observed points from the data distribution via the change of variables formula, as follows:

$$\log p_{\boldsymbol{\theta}}(\boldsymbol{x}) = \log q(f_{\boldsymbol{\theta}}^{-1}(\boldsymbol{x})) + \log \left|\det\left(\frac{\partial f_{\boldsymbol{\theta}}^{-1}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right)\right| = \log q(f_{\boldsymbol{\theta}}^{-1}(\boldsymbol{x})) + \sum_{i=1}^{K} \log \left|\det\left(\frac{\partial f_i^{-1}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right)\right|.$$

Finding a good approximation $p_{\boldsymbol{\theta}}$ is achieved through optimization of $\boldsymbol{\theta}$ so as to minimize the negative log likelihood of the observed dataset: $\mathcal{L}(\boldsymbol{\theta}) = -\frac{1}{N}\sum_{i=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})$. In practice, one will typically find the MLE $\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta})$ using some non-convex optimization method, such as stochastic gradient descent.

2.2 Differential Privacy

Differential privacy [15] has become the gold standard for ensuring the privacy of statistical analyses applied to sensitive databases. On a high level, it ensures that changing a single entry in the database will have only a small change in the distribution of analysis results.

Definition 1 ([15]) *A randomized algorithm $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ satisfies (ε, δ)-differential privacy (DP) if for any two input database $D, D' \in \mathcal{D}$ that differ in a single entry and for any subset of outputs $\mathcal{S} \subseteq \mathcal{R}$, it satisfies: $Pr[\mathcal{M}(D) \in \mathcal{S}] \leq e^{\epsilon} Pr[\mathcal{M}(D') \in \mathcal{S}] + \delta$.*

One common algorithmic approach for achieving differential privacy is adding noise that scales with the *sensitivity* of the function being evaluated, which is the maximum change in the function's value that can result from changing a single data point. Differentially private algorithms are robust to *post-processing*, meaning that any data-independent function of a differentially private output retains the same privacy guarantee, and they enjoy *composition*, meaning that the privacy parameters degrade gracefully as additional analyses are performed on the dataset. The simplest version of composition is that the privacy parameters ϵ and δ "add up" over multiple analyses.

Differentially Private Stochastic Gradient Descent was introduced in [1] as an approach to private non-convex optimization. At each step t, DP-SGD subsamples[1] a small set of data points and uses this batch to compute a gradient update. To achieve a differential privacy guarantee, DP-SGD achieves adds mean-zero Gaussian noise to the average of the per-example gradients. The standard deviation of this noise must scale with the sensitivity of the gradient estimation, although naturally this is unbounded by default, so the algorithm first clips each of the per-example gradients to ensure that each of their ℓ_2 norms is at most C, and then adds noise which scales with C.

[1] also introduced the *moments accountant*, which provided tight privacy composition across multiple gradient update steps in DP-SGD. To describe the moments accountant, given an algorithm $\mathcal{M}$ and two neighboring datasets D, D', first we denote the privacy loss of a particular outcome o as $L^{(o)} = \log(\Pr(\mathcal{M}_{\mathcal{D}} = o)/\Pr(\mathcal{M}_{\mathcal{D}'} = o))$. The moments accountant calculates a privacy budget by means of bounding the moments of the privacy loss random variable $L^{(o)}$. That is, if we consider the log of the moment generating function (MGF) of the privacy loss random variable evaluated at λ, i.e. $\alpha_{\mathcal{M}}(\lambda; \mathcal{D}, \mathcal{D}') = \log \mathbb{E}_{o\sim\mathcal{M}_{\mathcal{D}}}[e^{\lambda L^{(o)}}]$, the worst case over all neighboring databases $\max_{\mathcal{D},\mathcal{D}'} \alpha_{\mathcal{M}}(\lambda; \mathcal{D}, \mathcal{D}')$ composes linearly across multiple mechanisms (Theorem 2.1 [1]) and allows for conversion to an associated (ε, δ)-differential privacy guarantee through the relation $\delta = \min_\lambda \exp[\alpha_{\mathcal{M}}(\lambda) - \lambda\varepsilon]$.

Follow up work of [11] has provided an alternative analysis for DP-SGD utilizing privacy composition under the framework of *μ-Gaussian differential privacy*, which acts as the basis for our analysis. Noting that each iteration of DP-SGD achieves a μ-GDP guarantee depending on the standard deviation of noise applied to gradient updates, the overall privacy guarantee corresponding to k applications, each satisfying μ_i-GDP, is $\sqrt{\mu_1^2 + \mu_2^2 + \ldots \mu_k^2}$-GDP. One is then able to convert this overall μ-GDP guarantee to a corresponding (ε, δ)-differential privacy guarantee by noting that an algorithm is μ-GDP if and only if it is $(\varepsilon, \delta(\varepsilon))$-differentially private for all $\varepsilon \geq 0$, where $\delta(\varepsilon) = \Phi(-\frac{\varepsilon}{\mu} + \frac{\mu}{2}) - e^{\varepsilon}\Phi(-\frac{\varepsilon}{\mu} - \frac{\mu}{2})$ and $\Phi(\cdot)$ is the cumulative density function of the Normal distribution.

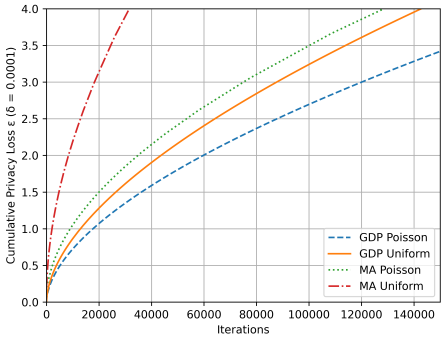

Figure 1: Cumulative privacy loss ε given Life Science training parameters ($q = b/N = 100/21384 = 4.676 \times 10^{-3}, \sigma = 2.1, \delta = 1 \times 10^{-4}$) as a function of training iterations.

3 Differentially Private Normalizing Flows

In this section we introduce our algorithm for differentially private density estimation via normalizing flows, DP-NF, presented in Algorithm 1 and based on the DP-SGD algorithm of [1], a differentially private method for performing stochastic gradient descent. We also briefly discuss performance improvements based on the data-dependent initialization of normalization layers and the use of a differentially private estimate of the distribution to act as a prior. We emphasize that our primary technical contribution is not in the design of these algorithms, but rather the novel application of these tools to the problem of differentially private density estimation in a way that yields substantial performance over prior work, as demonstrated by our empirical results in Section 4.

Training a normalizing flow model corresponds to minimizing the loss function $\mathcal{L}(\boldsymbol{\theta}) = -\frac{1}{N}\sum_{i=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})$. This loss function is non-convex when applied to the optimization of a non-linear normalizing flow model, and hence optimization is typically performed via gradient descent on $\boldsymbol{\theta}$. To make this training private in Algorithm 1, we update $\boldsymbol{\theta}$ using the DP-SGD algorithm of [1] described in Section 2.2, with some subtle yet important augmentations to the standard minibatch gradient descent process to allow for an explicit privacy guarantee, in accordance with DP-SGD.

[1]The original algorithm of [1] does this via Poisson subsampling, but can also be done via Uniform subsampling [40] and retain a privacy guarantee.

First, batches are sampled via uniform subsampling, i.e., sampled such that each possible batch of size b has equal likelihood of being chosen (as opposed to repeatedly shuffling the dataset and taking equally sized partitions of the dataset, which is often preferred in practice). Second, rather than computing the gradient with respect to the entire batch, the gradient with respect to each individual data point is calculated, clipped to have maximum ℓ_2 norm C, averaged, then added with a randomly sampled Gaussian noise vector.

Algorithm 1 also requires a *privacy accountant* to be specified as input. This privacy accountant will dynamically track the ε privacy loss incurred by composition over all gradient update steps as a function of the training parameters, and will halt the algorithm once a pre-specified budget is reached. Common choices for this accountant include the moments accountant (MA) [1] or composition via Gaussian differential privacy (GDP) [11]. In our experiments in Section 4, we yield preferable results using a GDP privacy accountant.

In summary, DP-NF in Algorithm 1 is a modified version of DP-SGD, instantiated to train a normalizing flow model with the analyst's choice of privacy accountant.

Algorithm 1 DP-NF, differentially private density estimation via normalizing flows

Input: Dataset $\boldsymbol{X} = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(n)}\}$, initialized parameters $\boldsymbol{\theta}$, learning rate η, batch size b, noise scale σ, upper-bound on ℓ_2 norm of per-example gradient C, training privacy budget ε, training privacy tolerance δ, privacy accountant P.
$t \leftarrow 1$
while $P(t, b/n, \sigma, C, \delta) < \varepsilon$ **do**
  Take a uniformly random subsample $I_t \subseteq \{1, \ldots, n\}$ with batch size b.
  for $i \in I_t$ **do**
    $\boldsymbol{g}_t^{(i)} \leftarrow \nabla_{\boldsymbol{\theta}} - \log p_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})$
    $\bar{\boldsymbol{g}}_t^{(i)} \leftarrow \boldsymbol{g}_t^{(i)} / \max\{1, ||\boldsymbol{g}_t^{(i)}||_2 / C\}$
  end for
  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta(\frac{1}{m}\sum_i \bar{\boldsymbol{g}}_t^{(i)} + \mathcal{N}(\boldsymbol{0}, \sigma^2 C^2 \boldsymbol{I}))$
  $t \leftarrow t + 1$
end while
Output $\boldsymbol{\theta}$

The privacy guarantees of DP-NF follow immediately from those of DP-SGD [1] when instantiated with the moments accountant, and from NoisySGD [5] when instantiated with the Gaussian differential privacy accountant.

In practice, one will find that many deep learning models (including the normalizing flow models used in our experiments) are much better optimized using adaptive learning rate optimization schemes. Given this, we found significant benefit in using a direct extension to DP-SGD which applies noisy gradients to the model according to the Adam [26] optimizer. Both methods achieve identical privacy guarantees given that computation of the first and second moments of the noisy gradients are merely deterministic data-independent functions of them. Thus they differ only in the post-processing of the noisy gradients, and the privacy guarantees are unchanged.

Differentially Private Data-Dependent Priors. Section 2.1 suggested the analyst choose a data-independent prior q, such as the multivariate spherical Gaussian. However, recent work suggests that modest improvements in empirical results can be achieved through the use of more complex priors, such as a mixture of Gaussians [36], or by fitting a Gaussian mixture model to the data [24]. A natural privacy-preserving approach would be to first use DP-EM [37] with privacy budget $(\varepsilon_1, \delta_1)$ to estimate a prior, and then refine the prior using DP-NF with privacy budget $(\varepsilon_2, \delta_2)$ to yield an encompassing normalizing flow model. This process would be $(\varepsilon_1 + \varepsilon_2, \delta_1 + \delta_2)$-differentially private, and may yield preferable results in contexts where the distribution at hand is highly discontinuous, but also locally nonlinear.

Data-Dependent Initialization of Normalization Layers. Intermediate normalization layers such as activation normalization [27] have been proposed as a means to improve the stability of normalizing flow models. Activation normalization is characterized by an offset and scaling of its inputs feature-wise by a learned set of parameters $\boldsymbol{b}$ and $\boldsymbol{w}$, i.e., $(\boldsymbol{x}^{(i)} - \boldsymbol{b})/\boldsymbol{w}$. In practice, these parameters are typically set via data-dependent initialization [38] by setting $\boldsymbol{b}$ and $\boldsymbol{w}$ as the per-feature means and

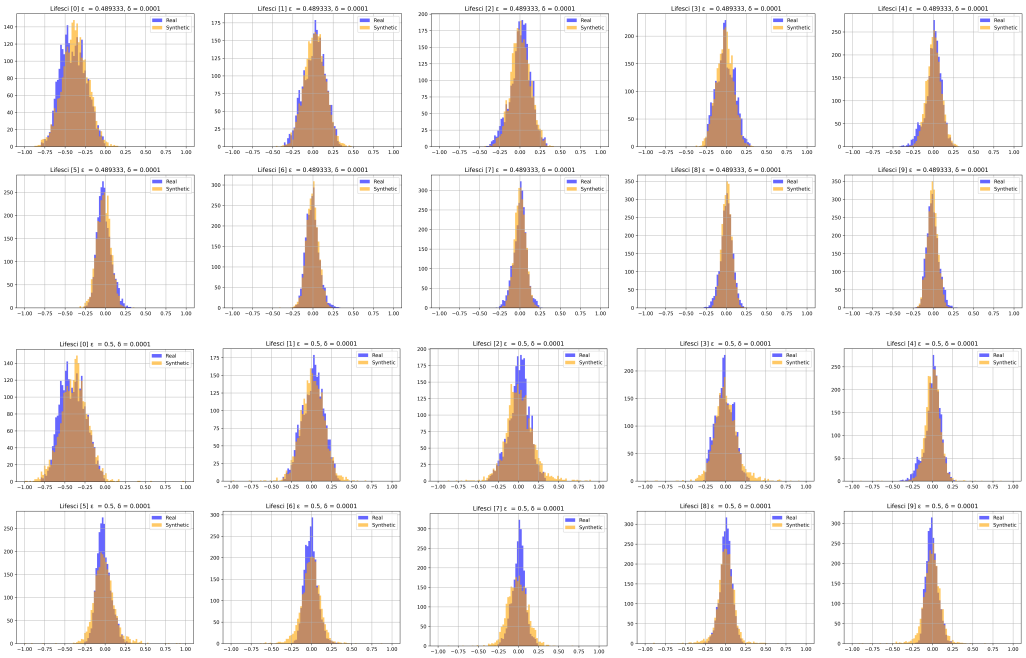

Figure 2: Dimension-wise histograms of synthetically generated Life Science data, superimposed over real data, for $\varepsilon = 0.5$ and $\delta = 10^{-4}$. **Top two rows:** DP-NF. **Bottom two rows:** DP-EM. Note that synthetic data from DP-NF represents the real data well, while DP-EM is relatively unable to to capture concentrated regions of density in the real data.

standard deviations observed throughout a forward pass of a sampled batch of data. These parameters can also be estimated privately, e.g., by applying the Laplace Mechanism [15] to the clipped mean and standard deviation, thus allowing for data-dependent initialization of these normalization layers.

4 Experimental Results

4.1 Datasets, Implementation, & Setup

The Life Science dataset is a standard density estimation benchmark dataset from the UCI machine learning repository [12] containing 26,733 real-valued records of dimension 10. This dataset was used in the original evaluation of our baseline model [37].

Experiments were run on a machine with 2 CPUs, 13 GB RAM, and a single NVIDIA Tesla K80 GPU, and took on the order of half an hour to five hours to run in wall-time, depending on the number of iterations and the dimensionality of the dataset. Models were implemented in the Jax [4] deep learning framework, and used privacy accounting implementations from TensorFlow Privacy [17].

Hyperparameter Search and Model Selection. Reported privacy budgets in our results correspond only to the training of each model, and does not include privacy loss from hyperparameter search and model selection. We chose not to selecting hyperparameters in a privacy-preserving manner because this was not the focus of our contribution and because it was not done in our baseline method.[2] It was generally observed that choices in network structure itself had relatively negligible impacts on results. We found that training parameters such as the gradient clipping bound and batch size had a much more substantial impact on model performance, which is consistent with observations made in [1].

[2]These can be done privately. For example, [21] provides discrete optimization methods that can be used for private hyperparameter search over discrete model architectures. [2] uses Report Noisy Max [16] for private model selection. Some work has also been done to account for high-performance models without having to spend a significant privacy budget [7, 29].

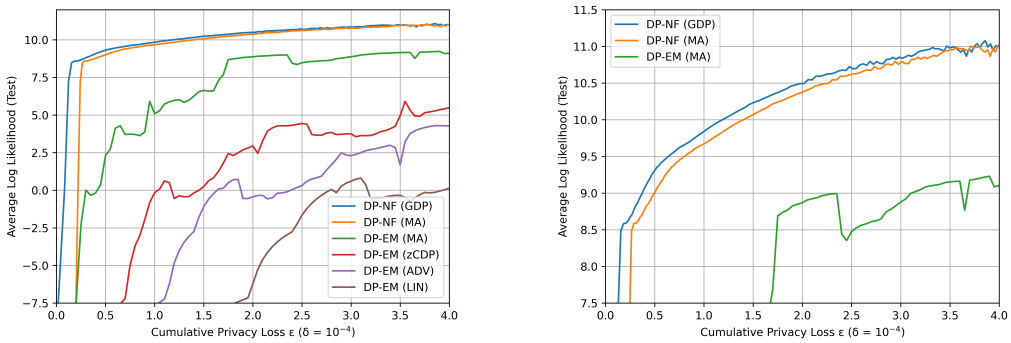

Figure 3: Average test log likelihood across ten independent cross-validation splits as a function of the cumulative privacy loss ε. **Left:** Reproduced version of Figure 3 [37] with the inclusion of DP-NF. **Right:** Figure cropped to region of peak performance. DP-EM was configured to use the Gaussian mechanism and with 3 components, as per the original work. DP-NF composed with GDP (as well as MA for fair comparison). δ fixed to 10^{-4}, approximately the inverse of the number of training examples in each split.

Table 1: Average test log likelihood for varying privacy budgets ε. Error bars denote standard deviation over ten independent cross-validation splits. Bolded results denote best performing model for a given ε.

Life Science $\delta = 1.00 \times 10^{-4}$	$\varepsilon = 0.50$	$\varepsilon = 1.00$	$\varepsilon = 2.00$	$\varepsilon = 4.00$
DP-NF (GDP)	$\mathbf{9.29 \pm 0.18}$	$\mathbf{9.83 \pm 0.12}$	$\mathbf{10.49 \pm 0.09}$	$\mathbf{11.01 \pm 0.24}$
DP-NF (MA)	8.99 ± 0.17	9.63 ± 0.12	10.37 ± 0.09	11.01 ± 0.18
DP-EM (MA)	1.96 ± 0.27	5.16 ± 0.20	8.67 ± 0.06	9.29 ± 0.06
DP-EM (zCDP)	-9.91 ± 0.49	-0.87 ± 0.37	2.51 ± 0.28	5.48 ± 0.18

Model Architecture. The architecture of the model used in our experiments was a variant of a Masked Autoregressive Flow (MAF) [36] composed of a repeated sequence of five blocks, each containing a MADE [19] layer, a reversal layer, and an optional activation normalization layer. Models were optimized via Adam, with default parameters of $\beta_1 = 0.9$ and $\beta_2 = 0.999$.

4.2 Density Estimation Tasks

We implemented our algorithm for differentially private normalizing flows on the Life Science dataset, and evaluated our performance against the baseline of DP-EM [37] for a variety of quantitative and qualitative metrics related to density estimation tasks.

First, Figure 2 shows that DP-NF provides a qualitative increase in sample quality under visualization. It presents dimension-wise histograms of synthetically generated features for all 10 features of the Life Science dataset, using DP-NF (top two rows) and DP-EM (bottom two rows) for comparison. Both methods used $\varepsilon = 0.5$ and $\delta = 10^{-4}$. In every plot, the synthetic data in orange is superimposed over the real data in blue. We qualitatively see that for nearly all ten features, the distribution of data generated by DP-NF closely matches that of the real data, while DP-EM was relatively unable to replicate regions of concentrated density for certain dimensions. This could be due to the fact that that for a fixed number of components, the DP-EM model is constrained to cover the support of the distribution and must ignore nuanced details. Normalizing flow models, on the other hand, have heightened expressiveness over traditional statistical methods like Gaussian mixture models, and we see that they are able to capture these nuances more readily.

Next we move to quantitative performance measures, and Figure 3 presents average log likelihood assigned to a held out test set under DP-NF and the baseline method DP-EM [37] as a function of ϵ. We divided the dataset into 10 pairs of training (90%) and test sets (10%), and reported the

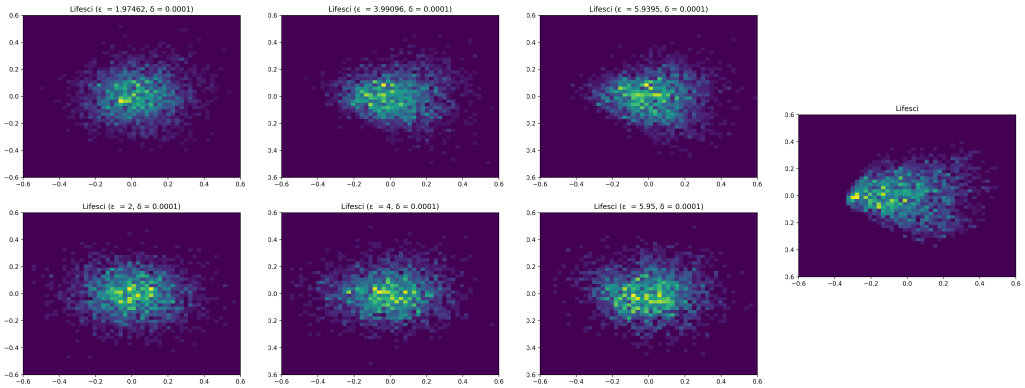

Figure 4: Synthetically generated Life Science data for ε = 2, 4, and 6, projected to two dimensions via PCA. **Top row:** DP-NF. **Bottom row:** DP-EM. **Right:** Real data. Note the compression to the left of the distribution of real data that is captured by DP-NF as ε increases, but not present in the synthetic data generated by DP-EM.

average test log likelihood per data point across the 10 independent trials. Better methods should assign higher log likelihood for points in the held out test set since these points were indeed sampled from the underlying distribution of interest. We found that DP-NF reliably assigned much higher likelihoods to holdout data than that of DP-EM for identical privacy budgets, across a variety of privacy accountant methods.

The privacy guarantees of DP-NF proved quite practical, providing substantial privacy improvements over DP-EM for the same model performance. For example, DP-NF matched the peak performance of DP-EM (achieved around $\varepsilon \approx 4$) for only an expenditure of $\varepsilon \approx 0.5$. These results are also listed in Table 1 with error bars showing standard deviation across 10 independent runs.

[37] showed performance of DP-EM under several different privacy accountant methods, with the moments accountant of [1] providing the best performance. We compared DP-NF using the moments accountant for fair comparison, and using the novel Gaussian differential privacy (GDP) accountant of [5]. Figure 3 and Table 1 show that DP-NF outperforms DP-EM for all privacy accountant methods considered for either model, emphasizing that while the GDP accountant does provide some benefit, the vasy majority of the performance improvements come from the DP-NF method itself.

As another qualitative evaluation of sample visualization, Figure 4 shows the density of synthetic data generated by each model when projected to two dimensional space via PCA, for varying ε values. The top row shows DP-NF, the bottom row shows DP-EM, and the right figure shows the real data. In all plots, lighter pixels correspond to regions of higher density, and dark pixels indicate lower density. We see that DP-NF is better able to capture some of the observable qualities exhibited in the real data, for example the gradual compression of density to the left of the distribution.

5 Conclusion

In this work, we have demonstrated the efficacy of differentially private normalizing flow models as a novel approach to the task of privacy-preserving density estimation. We have shown the ability of these models to assign high likelihoods to holdout data and generate qualitatively realistic synthetic data, improving on existing state-of-the-art methods. Going forward, there exist several interesting directions for further development. For example, it remains to be seen how normalization layers such as activation normalization, whose parameters are likely disproportionally sensitive to perturbation during differentially private optimization, could be better adapted to such. Further, in this study we only considered a particular subset of normalizing flows in existence. Although, many alternative neural density estimators capable of expressing highly discontinuous distributions are in continuous development, including FFJORD [20], Neural Spline Flows [13], Neural Autoregressive Flows [23], and Transformation Autoregressive Networks [34].

References

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security - CCS'16*, 2016.

[2] Brett K. Beaulieu-Jones, Zhiwei Steven Wu, Chris Williams, Ran Lee, Sanjeev P. Bhavnani, James Brian Byrd, and Casey S. Greene. Privacy-preserving generative deep neural networks support clinical data sharing. *bioRxiv*, 2018.

[3] Daniel Bittner, Anand Sarwate, and Rebecca Wright. *Using Noisy Binary Search for Differentially Private Anomaly Detection*, pages 20–37. 06 2018.

[4] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, and Skye Wanderman-Milne. JAX: composable transformations of Python+NumPy programs, 2018.

[5] Zhiqi Bu, Jinshuo Dong, Qi Long, and Weijie J. Su. Deep learning with gaussian differential privacy, 2019.

[6] Mark Bun and Thomas Steinke. Concentrated differential privacy: Simplifications, extensions, and lower bounds. *CoRR*, abs/1605.02065, 2016.

[7] Kamalika Chaudhuri and Staal A Vinterbo. A stability-based validation procedure for differentially private machine learning. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2652–2660. Curran Associates, Inc., 2013.

[8] Shuchi Chawla, Cynthia Dwork, Frank McSherry, and Kunal Talwar. On the utility of privacy-preserving histograms. In *UAI*, 2005.

[9] Laurent Dinh, David Krueger, and Yoshua Bengio. Nice: Non-linear independent components estimation. *CoRR*, abs/1410.8516, 2014.

[10] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp, 2016.

[11] Jinshuo Dong, Aaron Roth, and Weijie J. Su. Gaussian differential privacy. *CoRR*, abs/1905.02383, 2019.

[12] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017.

[13] Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows, 2019.

[14] Cynthia Dwork. Differential privacy: A survey of results. In *Theory and Applications of Models of Computation""TAMC*, volume 4978 of *Lecture Notes in Computer Science*, pages 1–19. Springer Verlag, April 2008.

[15] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the 3rd Conference on Theory of Cryptography*, TCC '06, pages 265–284, 2006.

[16] Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3–4):211–407, August 2014.

[17] Google et al. Tensorflow privacy, 2018.

[18] L. Fan and L. Xiong. Differentially private anomaly detection with a case study on epidemic outbreak detection. In *2013 IEEE 13th International Conference on Data Mining Workshops*, pages 833–840, 2013.

[19] Mathieu Germain, Karol Gregor, Iain Murray, and Hugo Larochelle. MADE: masked autoencoder for distribution estimation. *CoRR*, abs/1502.03509, 2015.

[20] Will Grathwohl, Ricky T. Q. Chen, Jesse Bettencourt, Ilya Sutskever, and David Duvenaud. FFJORD: free-form continuous dynamics for scalable reversible generative models. *CoRR*, abs/1810.01367, 2018.

[21] Anupam Gupta, Katrina Ligett, Frank McSherry, Aaron Roth, and Kunal Talwar. Differentially private approximation algorithms. *CoRR*, abs/0903.4510, 2009.

[22] Rob Hall, Alessandro Rinaldo, and Larry Wasserman. Differential privacy for functions and functional data. *J. Mach. Learn. Res.*, 14(1):703–727, February 2013.

[23] Chin-Wei Huang, David Krueger, Alexandre Lacoste, and Aaron C. Courville. Neural autoregressive flows. *CoRR*, abs/1804.00779, 2018.

[24] Pavel Izmailov, Polina Kirichenko, Marc Finzi, and Andrew Gordon Wilson. Semi-supervised learning with normalizing flows, 2019.

[25] Gautam Kamath, Or Sheffet, Vikrant Singhal, and Jonathan Ullman. Differentially private algorithms for learning mixtures of separated gaussians, 09 2019.

[26] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

[27] Diederik P. Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions, 2018.

[28] Diederik P. Kingma, Tim Salimans, and Max Welling. Improving variational inference with inverse autoregressive flow. *CoRR*, abs/1606.04934, 2016.

[29] Jingcheng Liu and Kunal Talwar. Private selection from private candidates. *CoRR*, abs/1811.07971, 2018.

[30] G. Mclachlan and K. Basford. Mixture models: Inference and applications to clustering, 01 1988.

[31] Ilya Mironov. Renyi differential privacy. *CoRR*, abs/1702.07476, 2017.

[32] Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. Smooth sensitivity and sampling in private data analysis. In *Proceedings of the Thirty-Ninth Annual ACM Symposium on Theory of Computing*, STOC '07, page 75–84, New York, NY, USA, 2007. Association for Computing Machinery.

[33] Rina Okada, Kazuto Fukuchi, Kazuya Kakizaki, and Jun Sakuma. Differentially private analysis of outliers, 2015.

[34] Junier B. Oliva, Avinava Dubey, Manzil Zaheer, Barnabás Póczos, Ruslan Salakhutdinov, Eric P. Xing, and Jeff Schneider. Transformation autoregressive networks, 2018.

[35] George Papamakarios. Neural density estimation and likelihood-free inference, 2019.

[36] George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 2338–2347. Curran Associates, Inc., 2017.

[37] Mijung Park, James Foulds, Kamalika Choudhary, and Max Welling. DP-EM: Differentially Private Expectation Maximization. In Aarti Singh and Jerry Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 896–904, Fort Lauderdale, FL, USA, 20–22 Apr 2017. PMLR.

[38] Tim Salimans and Diederik P. Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. *CoRR*, abs/1602.07868, 2016.

[39] H. S. Surendra and .S Mohan.H. A review of synthetic data generation methods for privacy preserving data publishing. *International Journal of Scientific Technology Research*, 6:95–101, 2017.

[40] Yu-Xiang Wang, Borja Balle, and Shiva Prasad Kasiviswanathan. Subsampled rényi differential privacy and analytical moments accountant. *CoRR*, abs/1808.00087, 2018.

[41] Yuncheng Wu, Yao Wu, Hui Peng, Juru Zeng, Hong Chen, and Cuiping Li. Differentially private density estimation via gaussian mixtures model. In *2016 IEEE/ACM 24th International Symposium on Quality of Service (IWQoS)*, pages 1–6, 2016.