Differentially Private Normalizing Flows for Privacy-Preserving Density Estimation

Chris Waites [1] **Rachel Cummings** [2]

Abstract

Normalizing flow models have risen as a popular solution to the problem of density estimation, enabling the ability to perform both exact probability density evaluation as well as high-quality synthetic data generation. However, in contexts where individuals are directly associated with the training data at hand, releasing such a model raises potential privacy concerns. In this work, we propose the use of normalizing flow models providing 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 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. Although, 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. Hence to the extent 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 (11) 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, given the strength of this definition, it acts as a compelling privacy notion to abide by in the design of privacy-preserving analyses.

Density estimators are a particularly strong class of analyses due to their versatile ability to address a wide range of tasks concerning a distribution, precisely why the existence of an accurate and privacy-preserving density estimator would be surprising. The private construction of such a model would implicitly yield a differentially private approach to anomaly detection—a task of substantial previous investigation (3; 26; 13)—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 (31) as it addresses many of the limitations imposed by the query model (10) by enabling 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 incur no additional cost to the overall privacy budget due to differential privacy's notion of immunity to post-processing.

Normalizing flow models present themselves as a particularly attractive approach to the task of density estimation due to their proven empirical ability to approximate highly complex 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 transformed distribution

[1]Department of Computer Science, Stanford University. [2]School of Industrial and Systems Engineering, Georgia Institute of Technology. Supported in part by NSF grants CNS-1850187 and CNS-1942772. Correspondence to: Rachel Cummings <rachelc@gatech.edu>.

Second workshop on *Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models* (ICML 2020), Virtual Conference

via an application of the change-of-variables formula. It has since been an open question to what degree normalizing flow models constructed in a differentially private manner could improve upon existing approaches to privacy-preserving density estimation.

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 2) that privately optimizes the model parameters via gradient descent according to DP-SGD (1). We apply this optimization to the parameters of a Masked Autoregressive Flow (28), our primary architecture of consideration, and achieve empirical results (Section 3) which outperform previous state-of-the-art approaches. Additionally, rather than performing composition via the moments accountant (MA) (1), we achieve tighter privacy guarantees via composition under the recently introduced notion of Gaussian differential privacy (8).

Details on related work are given in Appendix C.

2. Differentially Private Normalizing Flows

In the context of differentially private data analysis, the process by which we map an observed dataset to a set of trained model parameters is captured by the notion of a *randomized algorithm*. The goal in this case is to construct this randomized algorithm in such a way so as to satisfy a (ε, δ)-differential privacy guarantee, defined as follows:

Definition 1 ((11)) *A randomized algorithm $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ satisfies (ε, δ)-differential privacy (DP) if for any two input datasets $D, D' \in \mathcal{D}$ that differ in a single entry and for any subset of outputs $\mathcal{S} \subseteq \mathcal{R}$, it satisfies: $Pr[\mathcal{M}(D) \in \mathcal{S}] \leq e^{\epsilon} Pr[\mathcal{M}(D') \in \mathcal{S}] + \delta$.*

In a non-private context, one would typically perform this mapping by through some form of stochastic gradient descent, directly optimizing the model parameters to minimize the negative log likelihood of the observed data. Although naturally this approach does not yield an explicit privacy guarantee. To augment this procedure to produce such a guarantee, Differentially Private Stochastic Gradient Descent (DP-SGD), introduced in (1), offers itself as a technique for differentially private non-convex optimization. At each iteration, DP-SGD subsamples[1] a batch of data and computes the per-example gradient corresponding to each example in the batch. To achieve a differential privacy guarantee, DP-SGD places an upper-bound each of their ℓ_2 norms to be at most some constant C via gradient clipping. Then, the average of these per-example gradients is computed, added with mean-zero Gaussian noise exhibiting standard deviation proportional to C, and then applied to the model.

One can then go forward to analyze the privacy guarantees of multiple applications of DP-SGD across iterations via some privacy accounting scheme of choice. In recent years this has been primarily done through the moments accountant (1), expanded upon in Appendix B. Although, recent work (8) 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.

This technique for privacy-preserving optimization through DP-SGD, alongside the analysis facilitated via μ-GDP, is the basis for our approach. We present our approach for differentially private density estimation via normalizing flows, DP-NF, in Algorithm 1. 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 improvements over prior work, as demonstrated by our empirical results in Section 3.

2.1. Our Approach

Training a normalizing flow model corresponds to minimizing the loss function $\mathcal{L}(\boldsymbol{\theta}) = -\frac{1}{N}\sum_{i=1}^{N} \log p_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})$ through optimization of $\boldsymbol{\theta}$ via gradient descent. To make this training private in Algorithm 1, we update $\boldsymbol{\theta}$ using the DP-SGD algorithm of (1), with some subtle yet important augmentations to the standard minibatch gradient descent procedure to allow for an explicit privacy guarantee, in accordance with DP-SGD. 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 shuffling 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,

[1] The original algorithm of (1) does this via Poisson subsampling, but can also be done via uniform subsampling while retaining a privacy guarantee (32).

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
  $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \eta \cdot \frac{1}{b}(\sum_i \bar{\boldsymbol{g}}_t^{(i)} + \mathcal{N}(\boldsymbol{0}, \sigma^2 C^2 \boldsymbol{I}))$
  $t \leftarrow t + 1$
end while
Output $\boldsymbol{\theta}$

the 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 (1) or composition via Gaussian differential privacy (8). In our experiments in Section 3, we yield preferable results using a GDP privacy accountant.

In summary, DP-NF in Algorithm 1 is a direct instantiation of DP-SGD to train a normalizing flow model to minimize negative log likelihood, along with the analyst's choice of privacy accountant. The privacy guarantees of DP-NF follow immediately from those of DP-SGD (1) when instantiated with the moments accountant, and from NoisySGD (4) 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 optimizer (21). Note that both approaches yield identical privacy guarantees, given that computation of the first and second moments of the noisy gradients can be seen as merely a data-independent post-processing step.

Private Data-Dependent Priors. Previous normalizing flow literature has suggested that modest improvements in empirical results can be achieved through the use of more complex priors than the spherical Gaussian, such as a mixture of Gaussians (28) or a trained Gaussian mixture model (19). A natural privacy-preserving analog to the latter would be to fit a Gaussian mixture model via DP-EM (29) with privacy budget $(\varepsilon_1, \delta_1)$ to estimate a prior, and then refine this prior using DP-NF with privacy budget $(\varepsilon_2, \delta_2)$ to yield an encompassing normalizing flow model. This would be $(\varepsilon_1 + \varepsilon_2, \delta_1 + \delta_2)$-differentially private in the worst case, and could yield preferable results in contexts where the distribution at hand is highly discontinuous while exhibiting locally nonlinear density.

3. Experiments

Dataset. The Life Science dataset is a standard density estimation benchmark dataset from the UCI machine learning repository (9) containing 26,733 real-valued records of dimension 10. This dataset was used in the original evaluation of our baseline model (29).

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.[2] We chose not to select hyperparameters in a privacy-preserving manner as it was not done by our baseline and distracts from the focus of our contribution. Although, it was generally observed that changes in the network structure itself yielded negligible changes in results within reason. 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).

Model Architecture. The architecture of the model used in our experiments was a variant of a Masked Autoregressive Flow (MAF) (28) composed of a repeated sequence of five blocks, each containing a MADE (14) layer, a reversal layer, and an activation normalization layer. Models were optimized via Adam, with default parameters of $\beta_1 = 0.9$ and $\beta_2 = 0.999$.

3.1. Density Estimation Tasks

We implemented our algorithm for differentially private normalizing flows on the Life Science dataset, and evaluated

[2] We note that these can be done privately. For example, (15) provides discrete optimization methods that can be used for private hyperparameter search over discrete model architectures. (2) uses Report Noisy Max (12) for private model selection. Some work has also been done to account for high-performance models without having to spend a significant privacy budget (6; 23).

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)	8.49 ± 0.12	8.95 ± 0.15	9.63 ± 0.12	10.33 ± 0.08
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

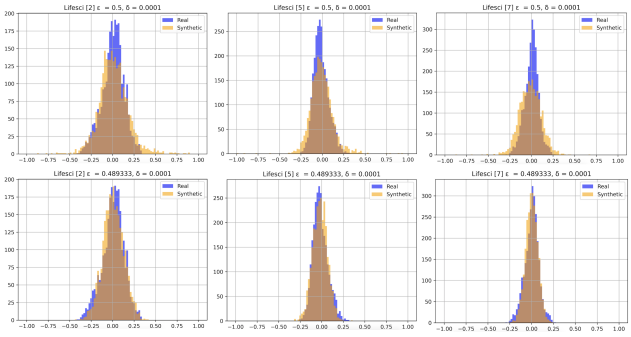

Figure 1. Dimension-wise histograms of synthetically generated Life Science data, superimposed over real data, for $\varepsilon = 0.5$ and $\delta = 10^{-4}$. **Top row:** DP-EM. **Bottom row:** DP-NF. One will note DP-NF's capability of capturing regions of concentrated density, whereas DP-EM struggles in this respect. Refer to Figure 4 for a holistic dimension-wise view of all features.

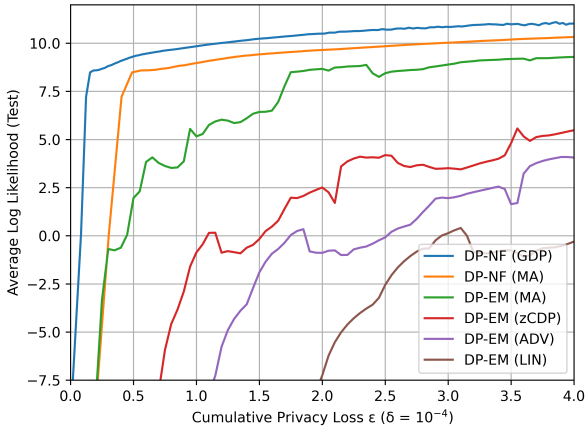

Figure 2. Average test log likelihood across ten independent cross-validation trials as a function of cumulative privacy loss ε (figure corresponds to Figure 3 of (29), with the inclusion of DP-NF). DP-EM configured with 3 mixture components and to use Gaussian mechanism, as per the original work. DP-NF composed with GDP, as well as MA for fair comparison. DP-NF outperforms DP-EM for all privacy accountant methods, even when both methods use the same technique (MA).

our performance against the baseline of DP-EM (29) for a variety of quantitative and qualitative metrics related to density estimation tasks.

First, Figure 1 (and more holistically, Figure 4) shows that DP-NF provides a qualitative increase in sample quality under visualization. It presents dimension-wise histograms of synthetically generated features of the Life Science dataset, using DP-NF and DP-EM 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 observe 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, DP-EM is constrained to cover the support of the distribution and must ignore nuanced details. Normalizing flow models, on the other hand, exhibit heightened expressiveness over traditional statistical methods like Gaussian mixture models, and we see that they are able to capture these nuances more readily.

Figure 2 presents average log likelihood assigned to a held out test set under DP-NF and the baseline method DP-EM (29) as a function of ϵ. We divided the dataset into 10 pairs of training (90%) and test sets (10%), and reported the average test log likelihood per data point across the 10 independent trials. We found that DP-NF reliably assigned higher likelihoods to holdout data than that of DP-EM for identical privacy budgets, across a variety of privacy accounting schemes. The privacy guarantees of DP-NF proved quite practical, matching 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. Figure 2 and Table 1 show that DP-NF outperforms DP-EM even controlling for the privacy accountant used, emphasizing that while the GDP accountant does provide some benefit, the underlying performance improvements truly come from the DP-NF method itself.

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

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

[6] 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.

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

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

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

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

[11] 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.

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

[13] 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.

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

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

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

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. *CoRR*, abs/1502.01852, 2015.

[18] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015.

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

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

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

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

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

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

[25] 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, pages 75–84, New York, NY, USA, 2007. Association for Computing Machinery.

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

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

[28] 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.

[29] 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.

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

[31] 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.

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

[33] 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.

A. Related Work

Gaussian mixture models (GMMs) are known to be a particularly strong density estimation baseline (27) 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 (24). 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 (25) 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 (20) 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 (33), 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 (29) 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) (5). 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 (7) 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 (16) 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.

B. Moments Accountant

The simplest version of differential privacy composition is that the privacy parameters ϵ and δ simply sum across multiple applications. Although to achieve a tighter bound, (1) also introduced the *moments accountant* which has been the standard approach to 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 datasets $\max_{\mathcal{D},\mathcal{D}'} \alpha_{\mathcal{M}}(\lambda; \mathcal{D}, \mathcal{D}')$ composes linearly across multiple mechanisms (Theorem 2.1 (1)) and allows for conversion to an associated (ε, δ)-differential privacy guarantee through the relation $\delta = \min_\lambda \exp[\alpha_{\mathcal{M}}(\lambda) - \lambda\varepsilon]$.

C. Additional Results

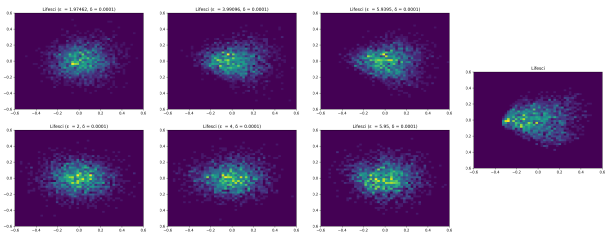

Figure 3. Synthetically generated Life Science data for ε = 2, 4, and 6, projected to two dimensions via PCA. **Top row:** DP-NF. **Bottom row:** DP-EM. **Right:** Real data. Note the compression to the left of the distribution of real data that is captured by DP-NF as ε increases, but not present in the synthetic data generated by DP-EM.

In addition to the previously provided figures, we also provide in Figure 4 a visualization of generated synthetic data projected from ten to two dimensional space using PCA for both models (DP-NF and DP-EM). Although significant information is lost when projecting down to a lower dimensional space, one may still observe broad similarities concerning the distribution learned by DP-NF over DP-EM in comparison to the real data.

D. Private Initialization of Normalization Layers

Intermediate normalization layers such as batch normalization (18) and activation normalization (22) have been shown to improve the stability of normalizing flow models. Although in our context, batch normalization is incompatible with our approach given that batch statistics are shared when computing the forward pass of the layer, precluding the ability to calculate truly independent per-example gradients as required by NoisySGD.

Although, activation normalization does not exhibit this limitation as no such batch statistics are calculated. Recall that activation normalization is characterized by an offset

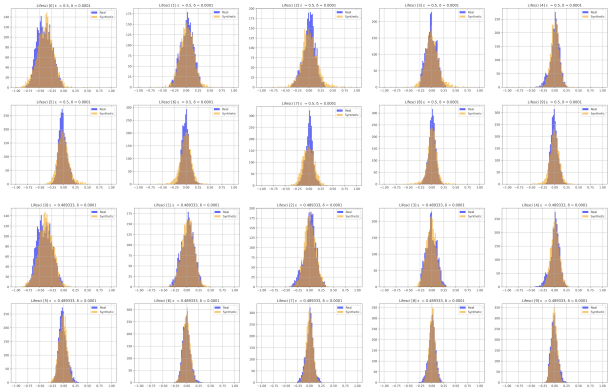

Figure 4. Dimension-wise histograms of synthetically generated Life Science data, superimposed over real data, for $\varepsilon = 0.5$ and $\delta = 10^{-4}$. **Top two rows:** DP-EM. **Bottom two rows:** DP-NF.

and scaling of its inputs feature-wise by a learned set of parameters $\boldsymbol{b}$ and $\boldsymbol{w}$, i.e. $\boldsymbol{y}^{(i)} \leftarrow (\boldsymbol{x}^{(i)} - \boldsymbol{b})/\boldsymbol{w}$. In practice, typically these parameters are set via data-dependent initialization (30) by computing a forward pass on a sampled batch of data and setting $\boldsymbol{b}$ and $\boldsymbol{w}$ to be the per-feature means and standard deviations of the inputs it had observed respectively.

Given that these statistics are not obfuscated in any manner, naturally this compromises privacy. A potential means to address this limitation is via a repeated application of noise when computing such statistics. This process is outlined in Algorithm 2, where $clip(\boldsymbol{X}, \tilde{c})$ clips the values of $\boldsymbol{X}$ to be in the range $[-\tilde{c}/2, \tilde{c}/2]$, $\mu(\boldsymbol{X})$ computes the feature-wise mean of $\boldsymbol{X}$, $\sigma(\boldsymbol{X})$ computes the feature-wise standard deviation of $\boldsymbol{X}$, and R is some data-independent parameter initialization method which maps standardized inputs to standardized outputs in expectation, e.g. He initialization (17).

Algorithm 2 DP-NF-INIT, data dependent initialization of activation normalization layers

Input: Dataset $\boldsymbol{X} = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(n)}\}$, transformation f (e.g. MADE (14)), number of layers K, initialization privacy budget ε, initialization privacy tolerance δ, data-independent parameter initialization method R (e.g. He initialization (17)).
$\{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K\} \leftarrow R()$
for $k = 1, \ldots, K$ **do**
  $\boldsymbol{X} \leftarrow clip(f_{\boldsymbol{\theta}^{(k)}}(\boldsymbol{X}), \tilde{c})$
  $\boldsymbol{b}^{(k)} \leftarrow \mu(\boldsymbol{X}) + Lap(\frac{2\sqrt{4K\ln(1/\delta)}\triangle\hat{\mu}}{\varepsilon})$
  $\boldsymbol{w}^{(k)} \leftarrow \sigma(\boldsymbol{X}) + Lap(\frac{2\sqrt{4K\ln(1/\delta)}\triangle\hat{\sigma}}{\varepsilon})$
  $\boldsymbol{X} \leftarrow (\boldsymbol{X} - \boldsymbol{b}^{(k)})/\boldsymbol{w}^{(k)}$
end for
Output concat $\boldsymbol{\theta}^{(1)}, \boldsymbol{b}^{(1)}, \boldsymbol{w}^{(1)}, \ldots, \boldsymbol{\theta}^{(K)}, \boldsymbol{b}^{(K)}, \boldsymbol{w}^{(K)}$

The privacy guarantee of this initialization scheme is (ε, δ)-differentially private as an immediate application of the Laplace mechanism paired with advanced composition (12).

Although the utility of activation normalization layers is quite evident, the original work (22) proposing such layers cited little evidence to support the idea that data-dependent initialization yielded statistically significant improvements over a default initialization scheme, i.e. $\boldsymbol{b} \leftarrow \mathbf{0}$ and $\boldsymbol{w} \leftarrow \mathbf{1}$. In our experimentation, we observed little distinction in contexts where the input data was assumed to be standardized and parameters were initialized to maintain variance between layers. Despite this, we include the approach for completeness for potential future contexts where data-dependent initialization of such parameters deems necessary.