Functional Rényi Differential Privacy for Generative Modeling

Dihong Jiang [1 2] **Sun Sun** [1 3] **Yaoliang Yu** [1 2]

Abstract

Recently, Rényi differential privacy (RDP) becomes an alternative to the ordinary differential privacy (DP) notion, for its convenient compositional rules and flexibility. However, existing mechanisms with RDP guarantees are based on randomizing a fixed, finite-dimensional vector output. In this work, following Hall et al. (2013) we further extend RDP to functional outputs, where the output space can be infinite-dimensional, and develop all necessary tools, e.g. (subsampled) Gaussian mechanism, composition, and post-processing rules, to facilitate its practical adoption. As an illustration, we apply functional RDP (f-RDP) to functions in the reproducing kernel Hilbert space (RKHS) to develop a differentially private generative model (DPGM), where training can be interpreted as releasing loss functions (in an RKHS) with RDP guarantees. Empirically, the new training paradigm achieves a significant improvement in privacy-utility trade-off compared to existing alternatives when $\epsilon = 0.2$.

1. Introduction

Differential privacy (DP, Dwork, 2006) becomes the defacto standard technique for releasing statistics of sensitive databases, which is designed to bound the output change of a randomized mechanism $\mathcal{M}$ given an incremental input deviation. Recently, Mironov (2017) generalizes DP to Rényi differential privacy (RDP) through α-Rényi divergence (Rényi, 1961), which shares many properties with the ordinary DP yet with easier composition analysis.

The popular mechanisms (e.g. Gaussian) towards DP or RDP essentially randomize a finite-dimensional vector output with noises. However, vector-based DP mechanisms are not readily amenable to *functions*, because (1) the dimension of a function can be infinite (e.g., kernel functions); (2) a function over a real-valued domain is characterized by infinitely many points (Hall et al., 2013), which makes it difficult to bound its sensitivity in the same way as the vector case in Dwork et al. (2014). Examples that require a functional DP mechanism include privately releasing the reward function in reinforcement learning (Wang & Hegde, 2019), the kernel function in kernel density estimation (Hall et al., 2013) and DPGM in this work.

Hall et al. (2013) made the most fundamental contribution to extending DP to functions. Essentially, the functional Gaussian mechanism is achieved by adding a sample path of Gaussian process to a function, in contrast to adding Gaussian noise to a vector. Evaluating the released DP function at arbitrarily many points (which will form a vector) will retain the same DP guarantee. However, there are no composition theorems and subsampled Gaussian mechanisms developed for functional DP in Hall et al. (2013), thus restricting its use in deep learning.

Due to the theoretical convenience and practical flexibility of RDP, in this work we aim to extend RDP to functions, with all necessary tools to facilitate its adoption in deep learning. Furthermore, we demonstrate its value via a particular application in DPGM where the loss function is in a RKHS. Our contributions can be summarized as:

- Theoretically, we develop the functional RDP, which is equipped with many useful tools including (subsampled) Gaussian mechanism, composition and post-processing theorems. We will show that functional RDP will share many important results with the vector-based variant.

- Empirically, with functional RDP, we propose a novel DPGM training paradigm by privatizing the loss function in RKHS, rather than truncating the RKHS to a finite-dimensional space and injecting Gaussian noise therein as in existing works. Our method is evaluated and compared across a wide variety of image datasets and DP guarantees, where our method consistently outperforms other baselines by a large margin. Notably, our method indicates better scalability at more stringent DP guaran-

[1]Cheriton School of Computer Science, University of Waterloo [2]Vector Institute [3]National Research Council. Correspondence to: Dihong Jiang <dihong.jiang@uwaterloo.ca>.

Workshop on Challenges in Deployable Generative AI at International Conference on Machine Learning (ICML), Honolulu, Hawaii, USA. 2023.

tees (e.g., $\epsilon = 1$ and 0.2), compared to state-of-the-art (SoTA) baselines.

2. RDP for functions (f-RDP)

For simplicity, we call ordinary DP (Dwork, 2006) v-RDP, ordinary RDP (Mironov, 2017) v-RDP, and functional DP (Hall et al., 2013) f-DP. Details are deferred to Appendix A. In this section, we aim to extend the definition of RDP to functional outputs, along with its associated calculus rules to facilitate practical adoption. In particular, we will show that the main results for v-RDP all extend to f-RDP.

2.1. Definition

Consider a class of functions over $T = \mathsf{R}^d$, i.e. $\{f_D : D \in \mathcal{D}\} \subseteq \mathsf{R}^T$. Analogous to Definition A.6, a weaker version of f-RDP based on cylinder sets is defined as follows:

Definition 2.1 ((α, ϵ)-RDP for functions, *weaker*)**.** Define cylinder sets $C_{S;B} = \{f \in \mathsf{R}^T : (f(\mathsf{x}_1), \ldots, f(\mathsf{x}_n)) \in B\}$, for all finite subsets $S = (\mathsf{x}_1, \ldots, \mathsf{x}_n)$ of T and Borel sets $B \subseteq \mathsf{R}^n$. Then, define $\mathcal{C}_S = \{C_{S;B} : B \in \mathcal{B}(\mathsf{R}^n)\}$ and $\mathcal{F}_0 = \bigcup_{S:|S|<1} \mathcal{C}_S$. We say the mechanism f_D^{f} satisfies (α, ϵ)-RDP over the field of cylinder sets, if for all adjacent inputs $D, D' \in \mathcal{D}$,

$$\Pr(f_D^{\mathsf{f}} \in A) \leq \exp(\epsilon) \Pr(f_{D'}^{\mathsf{g}} \in A)^{\frac{\alpha-1}{\alpha}}, \forall A \in \mathcal{F}_0. \tag{1}$$

Then, we give a stronger definition via α-Rényi divergence:

Definition 2.2 ((α, ϵ)-RDP for functions)**.** Denote the evaluation of function f_D^{f} on any finite subsets $S = (\mathsf{x}_1, \ldots, \mathsf{x}_n)$ of T by $\{f_D^{\mathsf{f}}(\mathsf{x}_1), \ldots, f_D^{\mathsf{f}}(\mathsf{x}_n)\} := f_D^{\mathsf{f}}(S)$. We say f_D^{f} is (α, ϵ)-RDP, if for all adjacent inputs $D, D' \in \mathcal{D}$, Rényi's α-divergence (of order $\alpha > 1$) between the distributions of $f_D^{\mathsf{f}}(S)$ and $f_{D'}^{\mathsf{g}}(S)$ satisfies:

$$D_\alpha(f_D^{\mathsf{f}}(S) \| f_{D'}^{\mathsf{g}}(S)) := \frac{1}{\alpha-1} \log \mathsf{E}_{x \sim q} \left(\frac{p(x)}{q(x)}\right)^\alpha \leq \epsilon, \tag{2}$$

where p, q are the density of $f_D^{\mathsf{f}}(S)$ and $f_{D'}^{\mathsf{g}}(S)$.

Remark 2.3. Definition 2.2 essentially claims that the distribution of any finite number of evaluations of function f_D^{f} and $f_{D'}^{\mathsf{g}}$ satisfies Definition A.2 (v-RDP).

Remark 2.4. Definition 2.2 implies Definition 2.1.

2.2. Post-processing theorem

As any data-independent post-processing preserves DP guarantee for v-RDP (Definition A.2), it also preserves DP guarantee for f-RDP with Remark 2.3. Specifically,

Theorem 2.5 (Post-processing theorem of f-RDP)**.** *If a function f_D is (α, ϵ)-RDP, so is $g \circ f_D$, where g is a post-processing mechanism that only depends on finite number of outputs of f_D.*

2.3. Conversion to (ϵ, δ)-DP

With Remark 2.4, we can use Definition 2.1 and reach reduction to Proposition 3 and its proof in Mironov (2017), by replacing the event from $f(D) \in S$ to $f_D^{\mathsf{f}} \in A$.

Proposition 2.6 (f-RDP conversion to f-DP)**.** *A function f_D^{f} that is (α, ϵ)-RDP is $(\epsilon + \frac{\log 1/\alpha}{\alpha - 1}, \delta)$-DP.*

2.4. Composition theorems

We first derive the parallel composition theorem of f-RDP:

Theorem 2.7 (Parallel composition of f-RDP)**.** *Given a partitioning function P, let $D_1, D_2, \ldots, D_m$ be the disjoint partitions by executing P on D. If function f_{D_i} is (α, ϵ_i)-RDP for $i = 1, 2, \ldots, m$, releasing $(f_{D_1}, \ldots, f_{D_m}) := f_D$ satisfies $(\alpha, \max_{i \in \{1,2,\ldots,m\}} \epsilon_i)$-RDP.*

Now we move on to the sequential composition theorem of f-RDP (extension of Theorem A.3 to functional mechanisms), which is important and required for composing the total privacy cost when we sample different sample paths from the Gaussian process over training iterations.

Theorem 2.8 (Sequential composition of f-RDP)**.** *Let $\{f_D : D \in \mathcal{D}\}$ and $\{g_D : D \in \mathcal{D}\}$ be two families of functions indexed by dataset D, where $f_D \in \mathcal{R}_1^T$ is (α, ϵ_1)-RDP and $g_D : \mathcal{R}_1^T \to \mathcal{R}_2^S$ is (α, ϵ_2)-RDP. Releasing the sequentially composed functional mechanism $h_D = (f_D, g_D \circ f_D) \in \mathcal{R}_1^T \times \mathcal{R}_2^S = (\mathcal{R}_1 \times \mathcal{R}_2)^{T^S}$ satisfies $(\alpha, \epsilon_1 + \epsilon_2)$-RDP.*

2.5. Gaussian mechanism

We also need to develop a Gaussian mechanism to retain f-RDP. For convenience, we first rewrite Proposition A.5 in a similar form to Proposition 3 in Hall et al. (2013) as:

Definition 2.9 (Gaussian mechanism for v-RDP, with non-isotropic Gaussian)**.** Let $M \in \mathsf{R}^{d \times d}$ be a positive definite symmetric matrix, the family of vectors $\{\mathsf{v}_D : D \in \mathcal{D}\} \in \mathsf{R}^d$ satisfies $\sup_{D, D'} \|M^{-\frac{1}{2}}(\mathsf{v}_D - \mathsf{v}_{D'})\|_2 \leq \Delta$ for all adjacent datasets $D, D' \in \mathcal{D}$. The Gaussian mechanism

$$\mathsf{vf}_D = \mathsf{v}_D + \sigma\Delta \cdot \mathcal{N}(0, M) \tag{3}$$

satisfies $(\alpha, \frac{\alpha}{2\sigma^2})$-RDP.

Now we define the Gaussian mechanism for f-RDP:

Proposition 2.10 (Gaussian mechanism for f-RDP)**.** *Let G be a sample path of a Gaussian process having mean zero and covariance function k. Let M denote the Gram matrix (as defined in Eq. (7)). Let $\{f_D : D \in \mathcal{D}\}$ be a family of*

functions indexed by database D. Releasing $f_D = f_D + G$ satisfies $(\ ; \frac{\ }{2\ ^2})$-RDP whenever Eq.(8) holds.

Particularly, when the function is in a RKHS, we have:

Corollary 2.11. For $\{f_D : D \in \mathcal{D}\} \subset \mathcal{H}$, releasing $f_D = f_D + G$ is $(\ ; \frac{\ }{2\ ^2})$-RDP (with respect to the cylinder $\ $-field) whenever $\sup_{D,D'} \|f_D - f_{D'}\|_{\mathcal{H}} \ $ and when G is a sample path of a Gaussian process with mean zero and covariance function k given by the kernel of $\mathcal{H}$.

2.6. Subsampled Gaussian mechanism (SGM)

Subsampling is a crucial component in existing DP deep learning algorithms (e.g., DP-SGD (Abadi et al., 2016)), which requires composing the privacy cost of $f(\cdot)$ for each subsampled batch over training iterations. This also applied to functional mechanisms when we subsample a batch from the whole dataset D to index the function f_S.

The current dominant python packages for DP, e.g., Tensorflow privacy and Opacus, apply v-RDP and compute the total $\ $ in three main steps: (1) compute the v-RDP guarantee for an SGM, based on a numerical procedure in Mironov et al. (2019); (2) sequentially compose v-RDP over training iterations; (3) convert v-RDP to v-DP. Since in previous sections we already showed that f-RDP shares the same results with v-RDP on steps (2) and (3), now we will show that an SGM for f-RDP can also be reduced to v-RDP as in Mironov et al. (2019), so the numerical procedure in Mironov et al. (2019) (and associated python packages) is amenable to f-RDP.

We apply the same subsampling strategy as in Mironov et al. (2019). Each element of the subsampled set S is independently drawn from D with probability q. SGM for f-RDP is given by $f_S = f_S + G$, where $\ $ and G are defined in Proposition 2.10. The reduction is immediate: for any finite set of points $V = \{x_1; \ldots; x_d\}$ (where $d < 1$), $f_S(V)$ will form a d-dimensional vector. Let $g_S(V) = M^{-1=2} f_S(V)$, we have $f_S(V) = g_S(V) + N(0; I_d)$, where $g_S(V)$ is a d-dimensional vector and $g_S(V)$ is the same SGM as in Mironov et al. (2019). Therefore, f-RDP shares the same guarantee of an SGM with v-RDP. Another intuition is that Mironov et al. (2019) reduce computing the $\ $-Rényi divergence of d-dimensional Gaussians to 1-dimensional Gaussians, which guides all of their subsequent derivations. When $d \to 1$, we can reach the same reduction from infinite-dimensional Gaussian (Gaussian process) to 1-dimensional Gaussian.

3. An application in DPGM

To demonstrate the empirical value of f-RDP, here we consider a particular example of training a DPGM through Maximum Mean Discrepancy (MMD). We defer the background to Appendix B.

3.1. Methodology

We use Gaussian kernel as our kernel function, i.e., $k(x; w) = \exp\{-\frac{\|x - w\|_2^2}{2h^2}\}$. Define

$$f_D = \frac{1}{N}\sum_{i=1}^{N} k(x_i; \cdot) = \frac{1}{N}\sum_{i=1}^{N} (x_i);$$

now we can rewrite Eq. (9) by plugging in f_D. Privatizing the terms relating to real data $x_i \in D$ $(i = 1; 2; \ldots; N)$ amounts to privatizing the function $f_D \in \mathcal{H}$ by Corollary 2.11, which leads to our private training objective:

$$\hat{L}_{MMD^2} = \frac{1}{N}\sum_{p=1}^{N} f_D(x_p) - \frac{2}{M}\sum_{j=1}^{M} f_D(w_j) + r; \quad (4)$$

where w is generated from a generative neural network $g_\ $ and $r = \frac{1}{M^2}\sum_{j=1}^{M}\sum_{q=1}^{M} k(w_j; w_q)$ is irrelevant to real data. Given the kernel is Gaussian, we can easily bound the sensitivity of f_D in RKHS norm. Wlog, assume $D; D'$ only differ in the last element, i.e. $x_N \neq x'_N$. Then,

$$\begin{aligned}(f_D - f_{D'}) &= \frac{1}{N}\sum_{i=1}^{N} (x_i) - \frac{1}{N}\sum_{i=1}^{N} (x'_i) \\ &= \frac{1}{N}\left((x_N) - (x'_N)\right)\end{aligned}$$

Since $k(x; y) = (x)^\top (y)$, we have

$$\begin{aligned}\|f_D - f_{D'}\|_{\mathcal{H}}^2 &= \frac{1}{N^2}\left[k(x_N; x_N) - 2k(x_N; x'_N) \right. \\ &\quad \left. + k(x'_N; x'_N)\right] \le \frac{2}{N^2} := \ ^2\end{aligned}$$

We follow the batch method in Hall et al. (2013) to release function f_D in practice, as it naturally fits the batch training manner, which amounts to sampling a path from the Gaussian process specified by any finite collection (batch) of points. Assuming the batch size is m, we concatenate $(x; w) = s$ (of size $2m$), for saving an additional privacy cost incurred by an additional sample path in each training iteration. Define $f_D(s) = (f_D(s_1); \ldots; f_D(s_{2m}))$, and M is a Gram matrix on s (similarly defined in Eq. (7)). Now we release $f_D(x)$ and $f_D(w)$ via $f_D(s)$ in Eq.(4) by:

$$f_D(s) \sim N(f_D(s); \ M): \quad (5)$$

We follow the approach in DP-MERF (Harder et al., 2021) to encode labels in the MMD loss (see Appendix C) for conditional generation.

3.2. Experiments

We evaluate our method both qualitatively and quantitatively on three image benchmarks. Implementation details are given in Appendix F.

Datasets: We consider widely used image benchmarks in related works, i.e. MNIST (LeCun et al., 1998), Fashion MNIST (Xiao et al., 2017), and CelebA (Liu et al., 2015). For MNIST and Fashion MNIST, we generate images conditioned on 10 respective labels. For CelebA, we condition on gender. See Appendix E for more details.

Evaluation metrics: We evaluate and compare DPGMs by two metrics via 60k generated images: (1) Fréchet Inception Distance (FID) (Heusel et al., 2017); (2) Classification accuracy. We train a convolutional neural network (CNN) as the classifier on generated images, then test the classifier on real images, where the performance is measured by the classification accuracy. We take 5 runs and report the average.

Baselines: (1) kernel-based methods: DP-MERF (Harder et al., 2021), DP-HP (Vinaroz et al., 2022), PEARL (Liew et al., 2022); (2) others: DP-CGAN (Torkzadehmahani et al., 2019), GS-WGAN (Chen et al., 2020), DP-Sinkhorn (Cao et al., 2021), G-PATE (Long et al., 2021), DataLens (Wang et al., 2021), DPDM (Dockhorn et al., 2022). All baselines are developed from v-RDP or v-DP.

Privacy regimes: Note that according to Definition A.1, the DP guarantee is weak when $\varepsilon = 10$, because $\exp(10) \approx 2.2 \times 10^4$, whereas the two probabilities are presumed to be comparable for practical deployment (e.g. $\varepsilon = 1$). However, a line of recent SoTA DPGMs only generate acceptable images at $\varepsilon = 10$ (Chen et al., 2020; Torkzadehmahani et al., 2019; Cao et al., 2021). Instead, we consider three values of ε, i.e., $10, 1, 0.2$, indicating three levels of DP guarantees. We put comparison under $(10, 10^{-5})$-DP in Appendix G.

3.3. Comparison with baselines

On MNIST and Fashion MNIST, while all baselines are able to generate reasonable images under $(10, 10^{-5})$-DP guarantee (see Appendix G), our method indicates more visual improvements for smaller ε with more diversity and less artifacts, as shown in Figures 1 and 2. The quantitative comparison is summarized in Table 1. Although DPDM is the only related work that is on a par with our method when $\varepsilon = 1$, our method significantly outperforms other baselines when $\varepsilon = 0.2$.

On a more complex colorful image dataset, i.e. CelebA,

Figure 1: Qualitative comparison under $(1, 10^{-5})$-DP on MNIST and Fashion MNIST

Figure 2: Qualitative comparison under $(0.2, 10^{-5})$-DP on MNIST and Fashion MNIST

Figure 3 shows that our method generates more diverse face images with identifiable gender attributes compared to baselines, which indicates its versatility and scalability.

Table 1: Quantitative comparison on MNIST and Fashion MNIST (FMNIST).

Method	ε	MNIST		FMNIST	
		FID ↓	Acc ↑	FID ↓	Acc ↑
DP-MERF	1	118.3	80.5	104.2	73.1
GS-WGAN	1	489.8	14.3	587.3	16.6
DP-HP	1	-	74.0	-	67.0
PEARL	1	121.0	78.2	109.0	68.3
G-PATE	1	153.4	58.8	214.8	58.1
DataLens	1	186.1	71.2	195.0	64.8
DPDM	1	23.4	93.4	37.8	73.6
Ours	1	29.5	93.4	49.5	78.8
DP-MERF	0.2	119.3	75.2	151.3	67.4
PEARL	0.2	133.0	77.6	160.0	68.0
G-PATE	0.2	-	22.0	-	18.0
DataLens	0.2	-	23.4	-	22.3
DPDM	0.2	61.9	71.9	78.4	57.0
Ours	0.2	26.5	91.3	46.2	78.4

(a) $\varepsilon = 1$ (b) $\varepsilon = 0.2$

Figure 3: The qualitative comparison on CelebA. $\delta = 10^{-5}$. Top row: female; bottom row: male. DPDM is unconditional.

4. Conclusion

We generalize RDP for vectors to functional mechanisms and develop all building blocks, e.g. (subsampled) Gaussian mechanisms, composition and post-processing theorems, to facilitate its adoption in deep learning. We show that those main results of v-RDP also hold for f-RDP. Equipped with f-RDP, we propose a novel approach for training a DPGM, by making the loss function in the RKHS private without truncating the RKHS feature map. Experimental results across different datasets and privacy costs indicate that our method (equipped with f-RDP and retaining the full discriminative capability of the kernel) consistently outperforms other kernel-based methods (with v-RDP) as well as non-kernel-based methods by a large margin. We expect our work to bridge the gap between RDP and functional mechanisms and enrich the family of DPGM.

References

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 308—318, 2016.

Balog, M., Tolstikhin, I., and Schölkopf, B. Differentially private database release via kernel mean embeddings. In *Proceedings of the 36th International Conference on Machine Learning*, 2018.

Cao, T., Bie, A., Vahdat, A., Fidler, S., and Kreis, K. Don't generate me: Training differentially private generative models with Sinkhorn divergence. In *Proceedings of the 34th Advances in Neural Information Processing Systems*, 2021.

Chen, D., Orekondy, T., and Fritz, M. GS-WGAN: A gradient-sanitized approach for learning differentially private generators. In *Proceedings of the 33rd Advances in Neural Information Processing Systems*, pp. 12673–12684, 2020.

Dockhorn, T., Cao, T., Vahdat, A., and Kreis, K. Differentially private diffusion models. arXiv:2210.09929, 2022.

Dwork, C. Differential privacy. In *Automata, Languages and Programming: 33rd International Colloquium, ICALP 2006, Venice, Italy, July 10-14, 2006, Proceedings, Part II 33*, pp. 1–12, 2006.

Dwork, C., Roth, A., et al. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3-4):211–407, 2014.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

Hall, R., Rinaldo, A., and Wasserman, L. Differential privacy for functions and functional data. *The Journal of Machine Learning Research*, 14(1):703–727, 2013.

Harder, F., Adamczewski, K., and Park, M. DP-MERF: Differentially private mean embeddings with random features for practical privacy-preserving data generation. In *International Conference on Artificial Intelligence and Statistics*, pp. 1819–1827, 2021.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Proceedings of 30th Advances in Neural Information Processing Systems*, volume 30, 2017.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998.

Li, C.-L., Chang, W.-C., Cheng, Y., Yang, Y., and Póczos, B. Mmd gan: Towards deeper understanding of moment matching network. In Proceedings of the 30th Advances in neural information processing systems, 2017.

Li, Y., Swersky, K., and Zemel, R. Generative moment matching networks. In International conference on machine learning, pp. 1718–1727, 2015.

Liew, S. P., Takahashi, T., and Ueno, M. PEARL: Data synthesis via private embeddings and adversarial reconstruction learning. In International Conference on Learning Representations, 2022.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In Proceedings of International Conference on Computer Vision (ICCV), 2015.

Long, Y., Wang, B., Yang, Z., Kailkhura, B., Zhang, A., Gunter, C. A., and Li, B. G-PATE: Scalable differentially private data generator via private aggregation of teacher discriminators. In Proceedings of 34th Advances in Neural Information Processing Systems, 2021.

McSherry, F. D. Privacy integrated queries: an extensible platform for privacy-preserving data analysis. In Proceedings of the 2009 ACM SIGMOD International Conference on Management of data, pp. 19–30, 2009.

Mironov, I. Rényi differential privacy. In 2017 IEEE 30th computer security foundations symposium (CSF) pp. 263–275. IEEE, 2017.

Mironov, I., Talwar, K., and Zhang, L. Rényi differential privacy of the sampled gaussian mechanism. 2019. arXiv:1908.10530.

Rahimi, A. and Recht, B. Random features for large-scale kernel machines. In Proceedings of the 20th International Conference on Neural Information Processing Systems, pp. 1177–1184, 2007.

Rényi, A. On measures of entropy and information. In Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, volume 1, pp. 547–562, 1961.

Sriperumbudur, B. K., Fukumizu, K., and Lanckriet, G. R. Universality, characteristic kernels and rkhs embedding of measures. Journal of Machine Learning Research, 12 (7), 2011.

Torkzadehmahani, R., Kairouz, P., and Paten, B. DP-CGAN: Differentially private synthetic data and label generation. In IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 98–104, 2019.

Van Erven, T. and Harremos, P. Rényi divergence and kullback-leibler divergence. IEEE Transactions on Information Theory, 60(7):3797–3820, 2014.

Vinaroz, M., Charusaie, M.-A., Harder, F., Adamczewski, K., and Park, M. J. Hermite polynomial features for private data generation. In Proceedings of the 39th International Conference on Machine Learning, volume 162, pp. 22300–22324, 2022.

Wang, B. and Hegde, N. Privacy-preserving q-learning with functional noise in continuous spaces. In Proceedings of the 32nd Advances in Neural Information Processing Systems, 2019.

Wang, B., Wu, F., Long, Y., Rimanic, L., Zhang, C., and Li, B. Datalens: Scalable privacy preserving training via gradient compression and aggregation. In Proceedings of the ACM SIGSAC Conference on Computer and Communications Security, pp. 2146–2168, 2021.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. arXiv:1708.07747, 2017.

A. Preliminary

In this section, we recall a few important related works in differential privacy.

A.1. Differential privacy for vectors (v-DP)

Definition A.1 ((ε, δ)-DP for vectors, (Dwork, 2006; Dwork et al., 2014))**.** A randomized mechanism $M : D \to R$ with domain D and range R satisfies (ε, δ)-differential privacy if for any two adjacent inputs $D, D' \in D$ and for any (measurable) subset of outputs $S \subseteq R$ it holds that

$$\Pr[M(D) \in S] \leq \exp(\varepsilon) \Pr[M(D') \in S] + \delta,$$

where adjacent inputs (a.k.a. neighbouring datasets) D, D' only differ in one entry. Particularly, when $\delta = 0$, we say that M is ε-DP.

A.2. Rényi differential privacy for vectors (v-RDP)

Mironov (2017) first formalizes Rényi differential privacy (RDP) which extends ordinary DP by using Rényi divergence (Rényi, 1961). RDP is shown to provide easier composition properties than the ordinary DP notion, and it can be easily converted to (ε, δ)-DP.

Definition A.2 ((α, ε)-RDP for vectors, (Mironov, 2017))**.** A randomised mechanism M is (α, ε)-RDP if for all adjacent inputs D, D', Rényi's α-divergence (of order $\alpha > 1$) between the distributions $M(D)$ and $M(D')$ satisfies:

$$D_\alpha(M(D) \| M(D')) := \frac{1}{\alpha - 1} \log \mathbb{E}_{x \sim q} \left(\frac{p(x)}{q(x)} \right)^\alpha \leq \varepsilon,$$

where p and q are the density of $M(D)$ and $M(D')$, respectively.

Conveniently, RDP is linearly composable:

Theorem A.3 (Sequential composition of v-RDP, (Mironov, 2017))**.** Let $f : D \to R_1$ be (α, ε_1)-RDP, $g : R_1 \times D \to R_2$ be (α, ε_2)-RDP, then running f, g sequentially to obtain $h : D \to R_1 \times R_2, h(D) := (f(D), g(f(D), D))$ satisfies $(\alpha, \varepsilon_1 + \varepsilon_2)$-RDP.

Similar to the parallel composition theorem for ordinary DP as in McSherry (2009), we complement the parallel composition for v-RDP:

Theorem A.4 (Parallel composition of v-RDP)**.** If mechanism M_i satisfies (α, ε_i)-RDP for $i = 1, 2, \ldots, m$, and let $D_1, D_2, \ldots, D_m$ be the disjoint partitions by executing a deterministic partitioning function from on D. Releasing $M_1(D_1), \ldots, M_m(D_m)$ satisfies $(\alpha, \max_{i \in \{1,2,\ldots,m\}} \varepsilon_i)$-RDP.

A.3. Gaussian mechanism for v-DP and v-RDP

Among multiple choices, the Gaussian mechanism is more suitable for the DP notion (where $\delta > 0$) and provides additional flexibility (e.g., sum of Gaussians is still a Gaussian). It is achieved by adding calibrated spherical Gaussian noise to a vector output.

Proposition A.5 (Gaussian mechanism for v-DP and v-RDP, (Dwork et al., 2014; Mironov, 2017))**.** Given a d-dimensional function $f : D \to R^d$. The Gaussian mechanism is given by:

$$M(D) = f(D) + \sigma \Delta_2 f \cdot \mathcal{N}(0, I_d),$$

where $\Delta_2 f = \max_{D, D' \in D; D \sim D'} \|f(D) - f(D')\|_2$. $M(D)$ is said to be: (1) (ε, δ)-DP if $\sigma = \sqrt{2 \ln(1.25/\delta)}/\varepsilon \cdot \Delta_2 f$ for $\varepsilon \in (0, 1)$, or (2) $(\alpha, \frac{\alpha}{2\sigma^2})$-RDP.

A.4. Differential privacy for functions (f-DP)

To our knowledge, Hall et al. (2013) first extended the DP notion to functional outputs:

Definition A.6 ((ε, δ)-DP for functions, (Hall et al., 2013))**.** Consider a class of functions indexed by database over $T = R^d$, i.e. $\{f_D : D \in D\} \subseteq R^T$. Define cylinder sets $C_{S,B} = \{f \in R^T : (f(x_1), \ldots, f(x_n)) \in B\}$, for all finite

subsets $S = (x_1, \ldots, x_n)$ of T and Borel sets $B \in \mathbb{R}^n$. Then, define $\mathcal{C}_S = \{C_{S,B} : B \in \mathcal{B}(\mathbb{R}^n)\}$ and $\mathcal{F}_0 = \bigcup_{S:|S|<\infty} \mathcal{C}_S$. We say the mechanism f_D satisfies (ε, δ)-DP over the field of cylinder sets, if for all $D, D' \in \mathcal{D}$:

$$\Pr[f_D \in A] \leq \exp(\varepsilon) \Pr[f_{D'} \in A] + \delta, \quad \forall A \in \mathcal{F}_0. \tag{6}$$

Proposition 5 in Hall et al. (2013) points out that whenever Eq. (6) holds, for any finite set of points $x_1, \ldots, x_n$ in T chosen a-priori, releasing the vector $[f_D(x_1), \ldots, f_D(x_n)]$ satisfies (ε, δ)-DP.

Gaussian mechanism for f-DP is reached by injecting calibrated Gaussian process into the function:

Proposition A.7 (Gaussian mechanism for f-DP, (Hall et al., 2013)). Let G be a sample path of a Gaussian process having mean zero and covariance function k. Let M denote the Gram matrix

$$M(x_1, \ldots, x_n) = \begin{pmatrix} k(x_1, x_1) & \cdots & k(x_1, x_n) \\ \vdots & \ddots & \vdots \\ k(x_n, x_1) & \cdots & k(x_n, x_n) \end{pmatrix}. \tag{7}$$

Let $\{f_D : D \in \mathcal{D}\}$ be a family of functions indexed by database D. Releasing $f_D = f_D + \sqrt{2\ln(1.25/\delta)}/\varepsilon \cdot G$ satisfies (ε, δ)-DP whenever

$$\sup_{D} \sup_{D' \in n(D)} \sup_{(x_1, \ldots, x_n) \in \mathcal{T}^n} \left\| M^{\frac{1}{2}}(x_1, \ldots, x_n) \begin{pmatrix} f_D(x_1) - f_{D'}(x_1) \\ \vdots \\ f_D(x_n) - f_{D'}(x_n) \end{pmatrix} \right\|_2 \leq 1. \tag{8}$$

Particularly, Hall et al. (2013) studied how to achieve (ε, δ)-DP for functions in a RKHS $\mathcal{H}$:

Corollary A.8 (Corollary 9 in (Hall et al., 2013)). For $\{f_D : D \in \mathcal{D}\} \subset \mathcal{H}$, releasing $f_D = f_D + \sqrt{2\ln(1.25/\delta)}/\varepsilon \cdot G$ is (ε, δ)-DP (with respect to the cylinder-field) whenever $\sup_{D,D' \in n(D)} \|f_D - f_{D'}\|_{\mathcal{H}} \leq 1$ and when G is a sample path of a Gaussian process with mean zero and covariance function k that is given by the reproducing kernel of $\mathcal{H}$.

B. Background and related works of DPGM

B.1. Background

Assuming a feature map $\phi : \mathcal{X} \to \mathcal{H}$, where $\mathcal{H}$ is a RKHS of some kernel, MMD (Gretton et al., 2012) is a non-parametric distance measure that compares two distributions p, q by:

$$\mathcal{L}_{\mathrm{MMD}^2}(p, q) = \|\mathbb{E}_{x \sim p}[\phi(x)] - \mathbb{E}_{w \sim q}[\phi(w)]\|_{\mathcal{H}}^2,$$

where $\mathbb{E}_{x \sim p}[\phi(x)] \in \mathcal{H}$ is also known as the (kernel) mean embedding (KME) of p.

Given a kernel $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, such that $k(x, w) = \langle \phi(x), \phi(w) \rangle_{\mathcal{H}}$, we can play the kernel trick to compute the (squared) MMD in an alternative way:

$$\mathcal{L}_{\mathrm{MMD}^2}(p, q) = \mathbb{E}_{x, x' \sim p} k(x, x') - 2\mathbb{E}_{x \sim p, w \sim q} k(x, w) + \mathbb{E}_{w, w' \sim q} k(w, w'),$$

which also implicitly lifts the KME into an infinite-dimensional space.

Sriperumbudur et al. (2011) suggest that if k is a characteristic kernel (e.g., Gaussian kernel), then MMD $= 0$ iff $p = q$, which makes MMD a practical tool in many applications, such as two-sample test (Gretton et al., 2012) and generative modeling (e.g., Li et al., 2015; 2017).

B.2. Related works

Balog et al. (2018) first proposed a DP database release mechanism via KME, by truncating the infinite-dimensional RKHS to a finite-dimensional feature space through random Fourier features and adding Gaussian noise to the mean of truncated feature embeddings of all data. This idea is further extended to generative modeling by DP-MERF (Harder et al.,

2021). Specifically, given the samples drawn from two distributions: $D = \{\mathbf{x}_i\}_{i=1}^N \sim p$ (true) and $W = \{\mathbf{w}_j\}_{j=1}^M \sim q$ (generated), empirical MMD with a kernel function k can be estimated by:

$$\mathcal{L}_{\text{MMD}^2}(p, q) = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{p=1}^{N} k(\mathbf{x}_i, \mathbf{x}_p) - \frac{2}{NM} \sum_{i=1}^{N} \sum_{j=1}^{M} k(\mathbf{x}_i, \mathbf{w}_j) + \frac{1}{M^2} \sum_{j=1}^{M} \sum_{q=1}^{M} k(\mathbf{w}_j, \mathbf{w}_q). \tag{9}$$

DP-MERF approximates the kernel by: $k(\mathbf{x}, \mathbf{w}) = \phi(\mathbf{x})^\top \phi(\mathbf{w})$, where $\phi(\mathbf{x}) \in \mathbb{R}^d$ and d is the feature dimension. The authors employ random Fourier features (Rahimi & Recht, 2007) as ϕ. Now the loss becomes: $\| \frac{1}{N} \sum_{i=1}^N \phi(\mathbf{x}_i) - \frac{1}{M} \sum_{j=1}^M \phi(\mathbf{w}_j) \|_2^2$.

Let $\mu_p = \frac{1}{N} \sum_{i=1}^N \phi(\mathbf{x}_i)$. Gaussian noise is added to μ_p to obtain $\mu_{\tilde{p}}$ with DP guarantees, which can be viewed as a privatized statistics of the whole real dataset. Thereafter, the training objective is to match the KME of generated data and $\mu_{\tilde{p}}$, without querying the real data any longer. A line of recent followup works, e.g., PEARL (Liew et al., 2022) and DP-HP (Vinaroz et al., 2022), boils down to improving the finite-dimensional truncation, and shows some further improvement in utility.

Compared to other DPGM approaches via DP-SGD (Abadi et al., 2016), DP-MERF is appealing in two aspects: (1) training efficiency, since noise is added to the KME of the whole database once and for all, whereas DP-SGD has to clip and perturb the gradient in each training iteration, which leads to significant training time overhead; (2) more scalable to smaller ϵ, e.g., $\epsilon = 1$ or below.

However, we observe that the generation by DP-MERF (and related followup methods) resembles "mean-like" images, which can be explained by their training objective, because matching the KME of all data is likely to lead to mode collapse. Moreover, truncating the RKHS into a finite dimensional space makes it easy to add Gaussian noise, but at the cost of potentially losing the fine ability to distinguish the data distribution from generation (any finite dimensional RKHS is not characteristic). Therefore, we turn to study the possibility of adding noise in the infinite-dimensional RKHS directly.

C. Extension to the conditional setting of our method

We follow the approach in DP-MERF to encode labels in the MMD loss. Consider a new kernel k as a product of two existing kernels: $k((\mathbf{x}, \mathbf{y}), (\tilde{\mathbf{x}}, \tilde{\mathbf{y}})) = k_{\mathbf{x}}(\mathbf{x}, \tilde{\mathbf{x}}) k_{\mathbf{y}}(\mathbf{y}, \tilde{\mathbf{y}})$, where we set $k_{\mathbf{x}}$ the same as the unconditional setting (i.e., Gaussian kernel) and $k_{\mathbf{y}}$ to be polynomial kernel of order-1, i.e., $k_{\mathbf{y}}(\mathbf{y}, \tilde{\mathbf{y}}) = \mathbf{y}^\top \tilde{\mathbf{y}}$. Now the function that we want to privately release becomes $f_D = \frac{1}{N} \sum_{i=1}^N k((\mathbf{x}_i, \mathbf{y}_i), (\cdot, \cdot))$. The sensitivity of f_D is the same as f_D. Thus, releasing f_D by *the batch method* is achieved by replacing f_D, k with f_D, k in Eq. (5).

D. Proofs

Theorem A.4 (Parallel composition of v-RDP)**.** *If mechanism $\mathcal{M}_i$ satisfies (α, ϵ_i)-RDP for $i = 1, 2, \ldots, m$, and let $D_1, D_2, \ldots, D_m$ be the disjoint partitions by executing a deterministic partitioning function P on D. Releasing $\mathcal{M}_1(D_1), \ldots, \mathcal{M}_m(D_m)$ satisfies $(\alpha, \max_{i \in \{1,2,\ldots,m\}} \epsilon_i)$-RDP.*

Proof. Without loss of generality, given two neighboring datasets D and D', assume that D contains one more element than D'. Executing f on D and D', we have partitions $D_1, D_2, \ldots, D_K$ and $D'_1, D'_2, \ldots, D'_K$, respectively. There exists j such that (1) D_j contains one more element than D'_j, and (2) $D_s = D'_s$ for $s = 1, 2, \ldots, K$ and $s \neq j$. Denote $\mathcal{M}_1(D_1), \ldots, \mathcal{M}_K(D_K)$ by $\mathcal{M}(D)$. Using additivity of Rényi divergence in (Van Erven & Harremos, 2014) (Thm 28):

$$\begin{aligned} D_\alpha(\mathcal{M}(D) || \mathcal{M}(D')) &= \sum_{i=1}^{K} D_\alpha(\mathcal{M}_i(D_i) || \mathcal{M}_i(D'_i)) \\ &= \sum_{i \in \{1,\ldots,K\}, i \neq j} D_\alpha(\mathcal{M}_i(D_i) || \mathcal{M}_i(D'_i)) + D_\alpha(\mathcal{M}_j(D_j) || \mathcal{M}_j(D'_j)) \\ &\leq \epsilon_j \leq \max_{i \in \{1,2,\ldots,K\}} \epsilon_i \end{aligned}$$

□

Proof. [of Remark 2.4] By taking logarithm and rearrangement from Eq. (1), we have

$$\frac{1}{\alpha-1}\log \quad \frac{\mathsf{P}(f_D^{\mathsf{f}} \in A)}{\mathsf{P}(f_{D'}^{\mathsf{g}} \in A)} \quad \mathsf{P}(f_{D'}^{\mathsf{g}} \in A) \quad \leq \epsilon \tag{10}$$

By the definition of cylinder sets, $f_D^{\mathsf{f}} \in A$ implies that $f_D^{\mathsf{f}}(S)$ is in some sets B for any finite subsets $S = (\mathsf{x}_1, \ldots, \mathsf{x}_n)$ of T. Thus,

$$\mathsf{P}(f_D^{\mathsf{f}} \in A) = \mathsf{P}(f_D^{\mathsf{f}}(S) \in B) = \int_S p(x)\,\mathrm{d}\mu(x)$$

where p is the density of $f_D^{\mathsf{f}}(S)$. Now we can translate Eq. (10) into

$$\frac{1}{\alpha-1}\log \quad \frac{\int_S p(x)\,\mathrm{d}\mu(x)}{\int_S q(x)\,\mathrm{d}\mu(x)} \quad \int_S q(x)\,\mathrm{d}\mu(x) \quad \leq \epsilon$$

Note that p, q are non-negative, so $\int (\frac{p(x)}{q(x)}) \; q(x)\,\mathrm{d}\mu(x) \geq \int_S (\frac{p(x)}{q(x)}) \; q(x)\,\mathrm{d}\mu(x)$. Compared to Eq. (2), it now suffices to show

$$\int_S (\frac{p(x)}{q(x)}) \; q(x)\,\mathrm{d}\mu(x) \geq \frac{\int_S p(x)\,\mathrm{d}\mu(x)}{\int_S q(x)\,\mathrm{d}\mu(x)} \quad \int_S q(x)\,\mathrm{d}\mu(x)$$

Define $p\;(x) = \frac{p(x)\mathbb{1}(x \in S)}{\int_S p(x)\,\mathrm{d}\;(x)}$, and $q\;$ is similarly defined. All we want to show reduces to

$$\int \frac{p\;(x)}{q\;(x)} \quad q\;(x)\,\mathrm{d}\mu(x) \geq 1$$
$$\Leftrightarrow \mathsf{E}_{x\;\sim q}[(\frac{p}{q})\;] \geq \mathsf{E}_{x\;\sim q}[\frac{p}{q}] \quad = 1$$

where the final step follows from Jensen's inequality. □

Theorem 2.5 (Post-processing theorem of f-RDP)**.** *If a function f_D is (α, ϵ)-RDP, so is $g \circ f_D$, where g is a post-processing mechanism that only depends on finite number of outputs of f_D.*

Proof. Given any finite subsets S, we reach the reduction of proof of the post-processing theorem in (Mironov, 2017):

$$\mathsf{D}\;(f_D(S)||f_{D'}(S)) \geq \mathsf{D}(g(f_D(S))||g(f_{D'}(S)))$$

□

Proposition D.1 (f-RDP conversion to f-DP)**.** *A function f_D^{f} that is (α, ϵ)-RDP is $(\epsilon + \frac{\log 1=}{1}, \delta)$-DP.*

Proof. Here we use Definition 2.1 (and associated notations) in this proof. To show that an (α, ϵ)-RDP function satisfies (ϵ', δ)-DP for functions, where $\epsilon' = \epsilon + \frac{\log 1=}{1}$, the objective becomes to show $\mathsf{P}[f_D^{\mathsf{f}} \in A] \leq \exp(\epsilon') \times \mathsf{P}[f_{D'}^{\mathsf{g}} \in A] + \delta$. By Definition 2.1, we have

$$\mathsf{P}(f_D^{\mathsf{f}} \in A) \leq \quad \exp(\epsilon)\mathsf{P}(f_{D'}^{\mathsf{g}} \in A) \quad ^{-1}, \quad \forall A \in \mathcal{F}_0$$

Denote $\mathsf{P}[f_{D'}^{\mathsf{g}} \in A]$ by Q,

- If $\exp(\epsilon)Q \leq \delta \;^{=(\quad 1)}$, then

$$\mathsf{P}(f_D^{\mathsf{f}} \in A) \leq \quad \exp(\epsilon)Q \quad ^{-1} \leq \delta \leq \exp(\epsilon')Q + \delta$$

- If $\exp(\epsilon)Q > \delta$ $^{=(\quad 1)}$, then

$$\begin{aligned}
\mathsf{P}(f\!\!\!f_D \in A) &\leq \quad \exp(\epsilon)Q \quad ^{\frac{\quad}{\quad 1}} \\
&= \exp(\epsilon)Q \quad \exp(\epsilon)Q \quad ^{\frac{1}{\quad}} \\
&\leq \exp(\epsilon)Q \cdot \delta \quad ^{\frac{\quad 1}{1}} \\
&= \exp(\epsilon + \frac{\log 1/\delta}{\alpha - 1})Q \\
&\leq \exp(\epsilon^0)Q + \delta
\end{aligned}$$

□

Theorem 2.7 (Parallel composition of f-RDP)**.** *Given a partitioning function P, let $D_1, D_2, \ldots, D_m$ be the disjoint partitions by executing P on D. If function f_{D_i} is (α, ϵ_i)-RDP for $i = 1, 2, \ldots, m$, releasing $(f_{D_1}, \ldots, f_{D_m}) := f_D$ satisfies $(\alpha, \max_{i \in \{1,2,\ldots,m\}} \epsilon_i)$-RDP.*

Proof. Given any finite subsets $S = (\mathsf{x}_1, \ldots, \mathsf{x}_n) \subset T$, we have

$$\begin{aligned}
\mathsf{D} \quad (f_D(S) || f_{D^0}(S)) &= \sum_{i=1}^{\not{K}} \mathsf{D} \quad (f_{D_i}(S) || f_{D_i^0}(S)) \\
&\leq \max_{j \in \{1,2,\ldots,K\}} \epsilon_j
\end{aligned}$$

□

Theorem 2.8 (Sequential composition of f-RDP)**.** *Let $\{f_D : D \in \mathcal{D}\}$ and $\{g_D : D \in \mathcal{D}\}$ be two families of functions indexed by dataset D, where $f_D \in \mathcal{R}_1^T$ is (α, ϵ_1)-RDP and $g_D : \mathcal{R}_1^T \to \mathcal{R}_2^S$ is (α, ϵ_2)-RDP. Releasing the sequentially composed functional mechanism $h_D = (f_D, g_D \circ f_D) \in \mathcal{R}_1^T \times \mathcal{R}_2^S = (\mathcal{R}_1 \times \mathcal{R}_2)^T \quad ^S$ satisfies $(\alpha, \epsilon_1 + \epsilon_2)$-RDP.*

Proof. According to Definition 2.2, our objective is to show for any finite subsets $X = (\mathsf{x}_1, \ldots, \mathsf{x}_n)$ of T and $Y = (\mathsf{y}_1, \ldots, \mathsf{y}_m)$ of S,

$$\begin{aligned}
&\mathsf{D} \quad (h_D(X,Y) || h_{D^0}(X,Y)) \leq \epsilon_1 + \epsilon_2 \\
\Leftrightarrow \mathsf{D} \quad &(f_D(X), g_D(f_D, Y) \;||\; f_{D^0}(X), g_{D^0}(f_{D^0}, Y) \quad \leq \epsilon_1 + \epsilon_2
\end{aligned}$$

Here we adapt the proof of Proposition 1 in (Mironov, 2017). Let F be the distribution of $f_D(X)$, G be the distribution of $g_D(f_D, Y)$, $H = (F, G)$, and F^0, G^0, H^0 are similarly defined on adjacent dataset D^0.

$$\begin{aligned}
\exp[(\alpha - 1)\mathsf{D} \quad (h_D(X,Y) || h_{D^0}(X,Y))] &= \int_{\mathsf{Z}^{R_1}} \int_{\mathsf{Z}^{R_2}} H(x,y) \quad H^0(x,y)^{\mathsf{1}} \quad \mathrm{d}x\,\mathrm{d}y \\
&= \int_{\mathsf{Z}^{R_1}} \int_{R_2} [F(x)G(x,y)] \quad [F^0(x)G^0(x,y)]^{\mathsf{1}} \quad \mathrm{d}x\,\mathrm{d}y \\
&= \int_{\mathsf{Z}^{R_1}} F(x) \quad F^0(x)^{\mathsf{1}} \int_{R_2} \mathsf{Z} \quad G(x,y) \quad G^0(x,y)^{\mathsf{1}} \quad \mathrm{d}y \quad \mathrm{d}x \\
&\leq \int_{R_1} F(x) \quad F^0(x)^{\mathsf{1}} \quad \cdot \exp[(\alpha - 1)\epsilon_2] \\
&\leq \exp[(\alpha - 1)(\epsilon_1 + \epsilon_2)]
\end{aligned}$$

□

Proposition D.2 (Gaussian mechanism for f-RDP)**.** *Let G be a sample path of a Gaussian process having mean zero and covariance function k. Let M denote the Gram matrix (as defined in Eq. (7)). Let $\{f_D : D \in \mathcal{D}\}$ be a family of functions indexed by database D. Releasing $f\!\!\!f_D = f_D + \sigma\Delta \cdot G$ satisfies $(\alpha, \frac{\quad}{2^{\,2}})$-RDP whenever Eq. (8) holds.*

Table 2: Training parameters for retaining DP on MNIST and Fashion MNIST. Both variants on CelebA use parameters in the row of Conditional. q is the subsampling rate, and σ is the noise multiplier.

	target ϵ	q	σ	epochs (iterations)
Conditional	10	0.001	0.60	200 (200k)
	1	0.001	1.95	200 (200k)
	0.2	0.001	8.0	200 (200k)
Parallel	10	0.01	1.0	200 (20k)
	1	0.01	5.75	200 (20k)
	0.2	0.01	25.0	200 (20k)

Proof. Consider any finite set $(\mathsf{x}_1, \ldots, \mathsf{x}_n) \in T^n$, the vector $(G(\mathsf{x}_1), \ldots, G(\mathsf{x}_n))$ follows a multivariate Gaussian with mean zero and covariance given by Eq. (7). Thus, evaluating f_D^{f} at any finite sets would form a vector that satisfies Eq. (3) in Definition 2.9, which completes the proof. □

E. Datasets

MNIST (LeCun et al., 1998) & Fashion MNIST (Xiao et al., 2017): MNIST contains hand-written digits images, whereas Fashion MNIST contains cloth and shoe images. Images in both datasets are single-channel, in the size of $1 \times 28 \times 28$, which are resized to $1 \times 32 \times 32$ and normalized to have 0.5 mean and 0.5 standard deviation. Both datasets have 10 classes. We adopt the official training and test split. MNIST and Fashion MNIST are made available under Creative Commons Attribution-Share Alike 3.0 license and MIT License, respectively.

CelebA (Liu et al., 2015): CelebA is a dataset including face images of celebrities. Each image is in the size of $3 \times 178 \times 218$ and has 40 binary attributes. All images are center-cropped to $3 \times 178 \times 178$, then resized to $3 \times 32 \times 32$, and normalized to have 0.5 mean and 0.5 standard deviation. We also adopt the official training, validation and test split, but randomly select 60k images from the training split as our training set. The CelebA dataset is available for non-commercial research purposes only, as described on their website.

F. Implementation

Generative network: Our unconditional generative network is based on the official *code* of MMD-GAN (Li et al., 2017). Encoding labels to the latent code leads to the conditional variant. We use the same network for all three datasets (only with different input channels). All networks are optimized by RMSprop with a learning rate 5×10^{-5}.

CNN classifier: We follow Cao et al. (2021) for the classifier implementation.

The CNN consists of following layers: Conv2d($input\_channels$, 32, kernel_size=3, stride = 2, padding=1) $\rightarrow$ Dropout(p=0.5) $\rightarrow$ ReLU $\rightarrow$ Conv2d(32, 64, kernel_size=3, stride = 2, padding=1) $\rightarrow$ Dropout(p=0.5) $\rightarrow$ ReLU $\rightarrow$ flatten $\rightarrow$ linear($flatten\_dim, output\_dim$) $\rightarrow$ Softmax.

The CNN classifier is optimized by Adam with default parameters. All classifiers are trained on synthetic data, and we report test accuracy on real test data as the evaluation metric.

Privacy: We use *Tensorflow privacy* for computing the total privacy cost, which only requires inputting a few important parameters, e.g. subsampling rate (or batch size), noise multiplier, training epochs (iterations), target δ ($\delta = 10^{-5}$ in all our experiments). We summarize the parameters in Table 2.

Baselines: We use the official code of DP-MERF to replicate their results under different ϵ and use the same evaluation metrics to add quantitative comparison. The numerical evaluations of rest baselines are cited from related works, as specified in the caption of the tables.

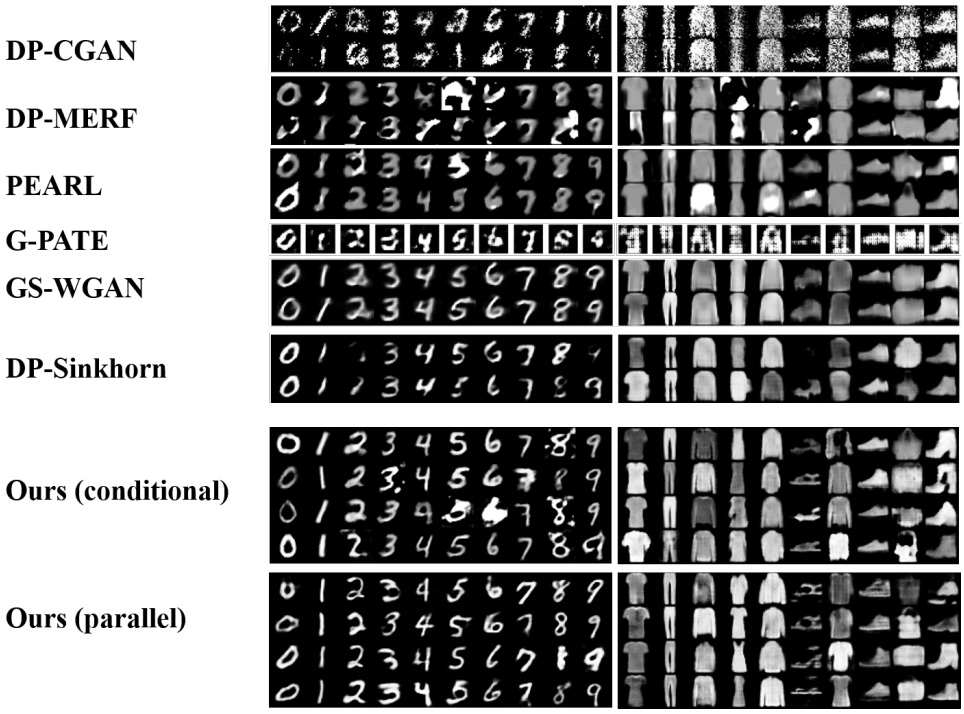

Figure 4: Qualitative comparison under $(10, 10^{-5})$-DP on MNIST and Fashion MNIST

Computation resources: All computation is conducted by one NVIDIA T4 GPU. It costs 1.5 hours to train a conditional generator on MNIST (and Fashion MNIST) and around 5 hours on CelebA. Each sub-model in the parallel variant takes only 10 minutes to train on MNIST (and Fashion MNIST) and 30 minutes on CelebA. A better GPU should compute faster. As a reference, DP-Sinkhorn and GS-WGAN can take up to 24 hours to train, and GS-WGAN even requires at least 2 GPUs (as they have 1000 discriminators).

G. Additional results

For completeness, we include the qualitative comparison between our method and related works under $(10, 10^{-5})$-DP on all three datasets in Figures 4 and 5. Numerical comparison is in Table 3. We note that $\epsilon = 10$ is usually considered a weak privacy regime. Generally, all baselines can generate decent images when $\epsilon = 10$, whereas our method still generates more diverse and more informative images, especially on CelebA.