Evaluating Generative Adversarial Networks on Explicitly Parameterized Distributions

Shayne O'Brien, Matt Groh, Abhimanyu Dubey
{shayneob, groh, dubeya}@mit.edu
Massachusetts Institute of Technology

Abstract

The true distribution parameterizations of commonly used image datasets are inaccessible. Rather than designing metrics for feature spaces with unknown characteristics, we propose to measure GAN performance by evaluating on explicitly parameterized, synthetic data distributions. As a case study, we examine the performance of 16 GAN variants on six multivariate distributions of varying dimensionalities and training set sizes. In this learning environment, we observe that: GANs exhibit similar performance trends across dimensionalities; learning depends on the underlying distribution and its complexity; the number of training samples can have a large impact on performance; evaluation and relative comparisons are metric-dependent; diverse sets of hyperparameters can produce a "best" result; and some GANs are more robust to hyperparameter changes than others. These observations both corroborate findings of previous GAN evaluation studies and make novel contributions regarding the relationship between size, complexity, and GAN performance.

1 Introduction

Generative adversarial network (GAN) optimization stability and convergence properties remain poorly understood despite the introduction of hundreds of GAN variants since their conception [8, 11]. While GAN learning and performance behavior has been studied [10, 19, 23], most existing work examining this relationship focuses on image datasets for which the underlying distribution parameterization is inaccessible [1, 2, 4, 18, 25]. This is problematic since claims of behavior that are made by modeling an unknown target distribution require a strong assumption for generalizability.

The goal of generative modeling is to approximate a distribution p_d by learning a parameterized distribution p_g, where both p_d and p_g are defined over samples. If we do not have full access to p_d, generalizability requires us to assume that the modeled dataset is a reasonable proxy for the family of distributions from which it was sampled. Without this assumption that is often only implicitly made, using images to understand GAN behavior limits conclusions to the data context being modeled.

We seek to address a gap in the literature by investigating GAN variant performance on datasets for which we have full access to the distribution parameterization. This allows us to study empirical performance on data where we can make claims of model behavior that generalize to the full distribution, as opposed to on image datasets for which this is not necessarily true. To this end, we examine the performance of 16 GAN variants on six explicitly parameterized multivariate distributions of four different dimensionalities and three different training set sizes.

Across 20 grid search trials, we observe that: (1) GANs exhibit similar performance trends across dimensionalities, (2) learning depends on the underlying distribution and its complexity, (3) the number of training samples can have a large impact on performance, (4) evaluation and relative comparisons

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

are metric-dependent, (5) diverse sets of hyperparameters can produce a "best" result, and (6) some GANs are more robust to hyperparameter changes than others. These findings corroborate those of previous GAN evaluation studies as well as contribute novel insights regarding the relationship between size, complexity, and GAN performance.[1]

2 Related Work

One notable work in this area by Lucic et al. [19] compares seven GAN variants in terms of modeling ability and optimization stability. The authors find that as computational budget increases, all tested models reach similar Frechét Inception Distance on the MNIST, Fashion-MNIST, CIFAR10, and CelebA datasets; and F1, precision, and recall on a synthetic dataset of convex polygons. They also discuss the difficulties of comparing GANs due to multiple valid ways to analyze performance.

Santurkar et al. [24] measure Inception Score and classification accuracy and report that the five GAN variants they train do not succeed at capturing distributional properties of the training set on the CelebA and LSUN datasets. The authors observe that the GAN distributions exhibit significantly less diversity at test time compared to the evaluation dataset, suggesting p_g is far from p_d.

In another study, Im et al. [13] evaluate GAN variant performance based on the original GAN criterion, least squares, maximum mean discrepancy, and improved Wasserstein distance. They show that for the three GAN variants they consider, test-time metrics do not favor networks that use the same training-time criterion on the MNIST, CIFAR10, LSUN, and Fashion-MNIST image datasets. The authors also examine performance as a function of sample size and show that some GANs exhibit faster performance increases than others as the number of training samples increases.

Lastly, Borji [4] provide a thorough discussion of the strengths and weaknesses of 26 quantitative and qualitative measures used for evaluating GANs trained on image datasets. They conclude that there is no single, best GAN evaluation measure. The authors suggest benchmarking models under identical architectures and computational budgets, and using more than a single metric to make comparisons.

3 Experimental Setup

In GANs, we define a prior probability distribution on input noise variables $p_z(\mathbf{z})$ and represent a mapping to the target data space $p_d(\mathbf{x})$ as $G(\mathbf{z}, \theta_G)$, where G is a fully differentiable neural network called the *generator* and θ_G are its parameters. We train G by simultaneously learning a fully differentiable network D, called the *discriminator* or *critic* and defined by $D(\mathbf{x}, \theta_D)$, that helps G during training. Whereas G is trained to mimic p_d, the learning objective, output, and precise task of D vary depending upon the GAN variant.

Models

As a case study, we examine the same seven GAN variants evaluated by Lucic et al. [19] and nine additional GAN variants that have been popularly discussed since their study was published. The primary difference between considered variants is whether the discriminator output can be interpreted as a probability (MMGAN, NSGAN [8], RaGAN [14], DRAGAN [16], FisherGAN [21], InfoGAN [5], ForwGAN, RevGAN, HellingerGAN, PearsonGAN, JSGAN [22]) or is unbounded (WGAN [1], WGANGP [9], LSGAN [20], BEGAN [3]). We summarize these models in Table 1.

In our implementations, both D and G consist of two feedforward network layers each; the full architecture has four layers total. We apply a ReLU activation function to the output of each layer and sample the noise prior $\mathbf{z}$ from $\mathcal{N}(0, \frac{h}{4}I)$, where h is the hidden dimension size. All models have the same number of trainable parameters except InfoGAN and BEGAN due to their use of latent variables as inputs to D and formation of D as an autoencoder, respectively. Chen et al. [5] argue that this difference is negligible for InfoGAN and we do not observe that it gives BEGAN any tangible advantage over other models. Trainable parameter counts can be found in Table 8.

[1] All code is publicly available at `https://github.com/shayneobrien/explicit-gan-eval`.

GAN Variant Loss Functions	
$\mathcal{L}^{\text{MMGAN}} = \mathbb{E}[\log(D(\mathbf{x}))] + \mathbb{E}[\log(1 - D(G(\mathbf{z})))]$	$\mathcal{L}^{\text{RaGAN}} = \mathbb{E}[\log(D(\mathbf{x}) - D(G(\mathbf{z})))] + \mathbb{E}[\log(1 - (D(G(\mathbf{z})) - D(\mathbf{x})))]$
$\mathcal{L}^{\text{NSGAN}} = \mathbb{E}[\log(D(\mathbf{x}))] - \mathbb{E}[\log(D(G(\mathbf{z})))]$	$\mathcal{L}^{\text{LSGAN}} = -\mathbb{E}[(D(\mathbf{x}) - 1)^2] + \mathbb{E}[D(G(\mathbf{z}))^2]$
$\mathcal{L}^{\text{WGAN}} = -\mathbb{E}[D(\mathbf{x})] + \mathbb{E}[D(G(\mathbf{z}))]$	$\mathcal{L}^{\text{BEGAN}} = \mathbb{E}[\lVert\mathbf{x} - D_{\text{AE}}(\mathbf{x})\rVert_1] - k_t\mathbb{E}[\lVert G(\mathbf{z}) - D_{\text{AE}}(G(\mathbf{z}))\rVert_1]$
$\mathcal{L}^{\text{WGANGP}} = \mathcal{L}^{\text{WGAN}} + \lambda\mathbb{E}\ [(\lVert\nabla_{\mathbf{z}}D(G(\mathbf{z}))\rVert_2 - 1)^2]$	$\mathcal{L}^{\text{DRAGAN}} = \mathcal{L}^{\text{MMGAN}} + \lambda\mathbb{E}[(\lVert\nabla_{\mathbf{x}}D(\mathbf{x} + \delta))\rVert_2 - 1)^2]]$
$\mathcal{L}^{\text{FisherGAN}} = \mathcal{L}^{\text{WGAN}} + \lambda(1 - \hat{\Omega}(D, G)) - \frac{\rho}{2}(\hat{\Omega}(D, G) - 1)$	$\mathcal{L}^{\text{InfoGAN}} = \mathcal{L}^{\text{MMGAN}}$ - $\lambda(\mathbb{E}[\log(Q(\mathbf{c}'\vert\mathbf{x}))])$
$\mathcal{L}^{\text{PearsonGAN}} = \mathbb{E}[D(\mathbf{x})] + \mathbb{E}[\frac{1}{4}D(G(\mathbf{z}))^2 + D(G(\mathbf{z}))]$	$\mathcal{L}^{\text{TVGAN}} = -\frac{1}{2}\mathbb{E}[\tanh(D(\mathbf{x}))] + \frac{1}{2}\mathbb{E}[\tanh(D(G(\mathbf{z})))]$
$\mathcal{L}^{\text{ForwGAN}} = \mathbb{E}[D(\mathbf{x})] + \mathbb{E}[\exp(D(G(\mathbf{z}))) - 1]$	$\mathcal{L}^{\text{RevGAN}} = \mathbb{E}[-\exp(D(\mathbf{x}))] + \mathbb{E}[-1 - (D(G(\mathbf{z})))]$
$\mathcal{L}^{\text{HellingerGAN}} = \mathbb{E}[1 - \exp(-D(\mathbf{x}))] + \mathbb{E}[\frac{1-\exp(D(G(\mathbf{z})))}{\exp(D(G(\mathbf{z})))}]$	$\mathcal{L}^{\text{JSGAN}} = \mathbb{E}[2 - (1 + \exp(-D(\mathbf{x})))] - \mathbb{E}[2 - \exp(D(G(\mathbf{z})))]$

Table 1: The loss function for each GAN variant with slight abuse of parameterization notation on the expectations, G, and D. Note that G is parameterized by θ_G, D is parameterized by θ_D, $x \sim p_d$, $z \sim p_g$, $\delta \sim \mathcal{N}(0, cI)$, $\hat{\Omega}(D, G) = \frac{1}{2}\mathbb{E}[D(\mathbf{x})]^2 - \frac{1}{2}\mathbb{E}[D(G(\mathbf{z}))]^2$, D_{AE} indicates that D is an autoencoder, $\mathbf{c} = [\mathbf{c_1}, \mathbf{c_2}]$ are structured latent variables where $\mathbf{c}'$ is sampled from the approximated distribution $p_{\mathbf{c}}(\mathbf{c}|\mathbf{x})$, $\nabla_{(\cdot)}$ is the gradient of the loss with respect to $(\cdot)$, and k_t, λ, and ρ are introduced hyperparameters.

Data

We train each of these variants by randomly sampling 1,000, 10,000, and 100,000 data points from the following six explicitly parameterized multivariate distributions: Gaussian with mean μ and symmetric, full rank covariance $\mathbf{\Sigma}$ both from $[0, 1]$; exponential with inverse mean shape λ from $[0, 1]$; beta with shape parameters a and β both from $[0, 1]$; gamma with shape k from $[0, 10]$ and scale θ from $[0, 2]$; Gumbel with location μ and scale β both from $[0, 1]$; and Laplace with location μ and scale β both from $[0, 1]$. For each of these distributions and numbers of samples, we generate datasets of 16, 32, 64, and 128 dimensions. We note that by the Universal Approximation Theorem, our proposed network architecture should be able to model each of these distributions without exception.

Hyperparameters

For all models and data distributions, we conduct 20 grid search trials with random network initializations for learning rates $\gamma \in [2\mathrm{e}^{-1}, 2\mathrm{e}^{-2}, 2\mathrm{e}^{-3}]$, hidden dimension sizes $h \in [32, 64, 128, 256, 512]$, and batch size $b = 1024$.[2] For models with introduced hyperparameters, we use those given in the original the paper. We use the Adam optimizer with default settings [15] and train for 25 epochs.

Measuring Divergence

We evaluate the difference between p_d and p_g using Kullback-Leibler divergence (KL), Jensen-Shannon divergence (JS), and Wasserstein Distance (WD). KL and JS focus on the alignment of the modes of the distributions and WD emphasizes how much p_g must be modified to reach p_d. Whereas JS and WD are symmetric, KL is not. For any of these measures, a value of 0 can be interpreted as indicating the two distributions being compared are identical [6, 17, 26]. We report results as the divergence between a generated batch and a test batch of size $b = 1024$ at the end of every epoch.

Estimating p_g

Although we have access to the true data distribution p_d, we must estimate the probability distribution of p_g. Since the data dimensionality is low, we construct a dimension-wise histogram for each data point.[3] In doing so, we assume that each dimension is independent from the others. This assumption is valid in the case of all experiments involving non-Gaussian data, which follows from the multivariate model being a product of the marginal distributions. To select the optimal bin width B_w in the histogram, we follow the Freedman-Diaconis rule: $B_w = \frac{2 \cdot IQR(\tilde{\mathbf{x}})}{\sqrt[3]{M}}$, where IQR is the

[2] We also ran full experiments for $b \in [128, 256, 512]$, but limit our analyses to b = 1024 as results across different batch sizes are not comparable due to greater noise in the data generation process at lower values of b.

[3] Kernel density estimation was found to give similar outputs while being more computationally expensive.

inter-quartile range of the M samples $\tilde{\mathbf{x}} = \{x_1, ..., x_M\}$ from the distribution being approximated. This initialization minimizes the difference between the areas under the empirical and theoretical probability distributions [7].

4 Results

In our analyses, we take the same approach as Lucic et al. [19] and Im et al. [13]: we let the "best" hyperparameter setting be the one that achieved the lowest minimum performance on average across all trials for each distribution, metric, and number of training samples, respectively. We include results, visualizations, and evidence to support all conclusions in the appendices. For the best hyperparameter settings in our learning environment, we find that:

1. **GANs exhibit similar learning trends across dimensionalities**: For many of the models, performance under the best hyperparameter setting consistently follows a trend across dimensionalities for all three tested metrics. At the same time, performance generally worsens with increased dimensionality. See Figures 1, 2, 3.
2. **Learning depends on the underlying distribution and its complexity:** Models which do well on some distributions perform poorly on the same distribution with higher dimensionality, or on other distributions of the same dimensionality. It is not immediately apparent that these differences are due to model design. Lucic et al. [19] make a similar finding in the case of image datasets with varying complexities. See Figures 1, 2, 3.
3. **Number of training samples can have a large impact on performance:** Some GAN variants are able to achieve the same performance learning from 1,000 samples as 10,000 or 100,000 samples, while others show large performance jumps with increased amounts of data. At the same time, almost all GAN variants begin to worsen in performance within five epochs for 1,000 training samples. The number of training samples seems to be critical to some models' performances, which was also noted by Im et al. [13]. See Figures 4, 5, and 6.
4. **Evaluation and comparison are metric-dependent:** Relative ranking of GAN variants according to performance varies depending on the evaluation metric used to rank them. No single GAN performed best across all metrics for any dataset or dimensionality. We concur with previous studies that GANs generally perform the same, although there are variants that perform worse than others on some distributions [4, 12, 13, 19, 24]. We warn against ranking models as relative differences can be marginal. See Tables 2, 3, 4, and 5.
5. **Diverse sets of hyperparameters can produce a "best" result:** Many, diverse hyperparameter settings yielded superior performances to the best average minimum performance, but these models did not achieve those minima with tight confidence bounds. Furthermore, we see that even on the best performing hyperparameter settings, our tested models preferred widely different hidden dimensionalities and learning rates; some variants with less parameters outperformed others that had more. We agree with previous work that it is important to present results that are able to be consistently reproduced [4, 13, 19, 18, 25]. See Table 6.
6. **Some GANs are more robust to hyperparameter changes than others:** With respect to the distribution, dimensionality, and training set size being approximated, some models yielded average minimum performances for more hyperparameters than others that fell within the confidence interval of the best average minimum performance under consideration. This is an indication that some GANs can perform well under a greater range of hyperparameter settings than others. See Table 7.

5 Future Work

In future work, we plan to analyze cases where GAN variants underperform relative to others and relate the characteristics of the distribution being modeled to the assumptions made in designing the variant, e.g. by empirically considering whether a normally distributed prior hurts performance on non-normal distributions. We would also like to use longer training times and more complex models to evaluate additional synthetic datasets such as multivariate mixture models, colored circles, and autoencoded image datasets.

Acknowledgements

We would like to thank Deb Roy, Iyad Rahwan, Manolis Zampetakis, and the Media Lab Consortium for their support of this research. Aleksander Mądry and Costis Daskalakis provided us with inspiration to pursue this project. Shayne O'Brien is partially funded through a National Science Foundation Graduate Research Fellowship.

References

[1] Martín Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pages 214–223, 2017.

[2] Sanjeev Arora, Rong Ge, Yingyu Liang, Tengyu Ma, and Yi Zhang. Generalization and equilibrium in generative adversarial nets (GANs). In *Proceedings of the 34th International Conference on Machine Learning*, pages 224–232, 2017.

[3] David Bethelot, Thomas Schumm, and Luke Metz. BEGAN: Boundary equilibrium generative adversarial networks. *arXiv preprint arXiv:1703.10717*, 2017.

[4] Ali Borji. Pros and cons of GAN evaluation measures. *CoRR*, abs/1802.03446, 2018. URL `http://arxiv.org/abs/1802.03446`.

[5] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. InfoGAN: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2172–2180, 2016.

[6] Dominik Endres and Johannes Schindelin. A new metric for probability distributions. In *IEEE Transactions on Information Theory*, pages 1858–1860, 2003.

[7] David Freedman and Persi Diaconis. On the histogram as a density estimator: L2 theory. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, pages 453–476, 1981.

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[9] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein GANs. In *Advances in Neural Information Processing Systems*, pages 5767–5777. 2017.

[10] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, Günter Klambauer, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a Nash equilibrium. *arXiv preprint arXiv:1706.08500*, 2017.

[11] Avinash Hindupur. The GAN zoo. *GitHub Repository*, 2017. URL `https://github.com/hindupuravinash/the-gan-zoo`.

[12] Gao Huang, Yang Yuan, Qiantong Xu, Chuan Guo, Yu Sun, Felix Wu, and Kilian Weinberger. An empirical study on evaluation metrics of generative adversarial networks. 2018. URL `https://openreview.net/forum?id=Sy1f0e-R-`.

[13] Daniel Jiwoong Im, Alllan He Ma, Graham W. Taylor, and Kristin Branson. Quantitatively evaluating GANs with divergences proposed for training. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=SJQHjzZ0-`.

[14] Alexia Jolicoeur-Martineau. The relativistic discriminator: A key element missing from standard GAN. *arXiv preprint arXiv:1807.00734*, 2018.

[15] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

[16] Naveen Kodali, Jacob Abernethy, James Hays, and Kira Zsolt. On convergence and stability of GANs. *arXiv preprint arXiv:1705.07215*, 2017.

[17] Soloman Kullback and Richard Leibler. On information and sufficiency. pages 79–86, 1951.

[18] Holden Lee, Rong Ge, Tengyu Ma, Andrej Risteski, and Sanjeev Arora. On the ability of neural nets to express distributions. In *Journal of Machine Learning Research*, pages 1–26, 2017.

[19] Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are GANs created equal? A large-scale study. *arXiv preprint arXiv:1711.10337*, 2017.

[20] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2813–2821, 2017.

[21] Youssef Mroueh and Tom Sercu. Fisher GAN. In *Advances in Neural Information Processing Systems*, pages 3310–3320, 2017.

[22] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-GAN: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems*, pages 271–279. 2016.

[23] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *Advances in Neural Information Processing Systems*, pages 2234–2242, 2016.

[24] Shibani Santurkar, Ludwig Schmidt, and Aleksander Madry. A classification-based perspective on GAN distributions. *arXiv preprint arXiv: 1711.00970*, 2017.

[25] Lucas Theis, Aäron van den Oord, and Matthias Bethge. A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844*, 2015.

[26] Cédric Villani. *Optimal Transport: Old and New*, volume 338. Springer Science & Business Media, 2008.

A GANs exhibit similar learning trends across dimensionalities, and learning depends on the underlying distribution and its complexity

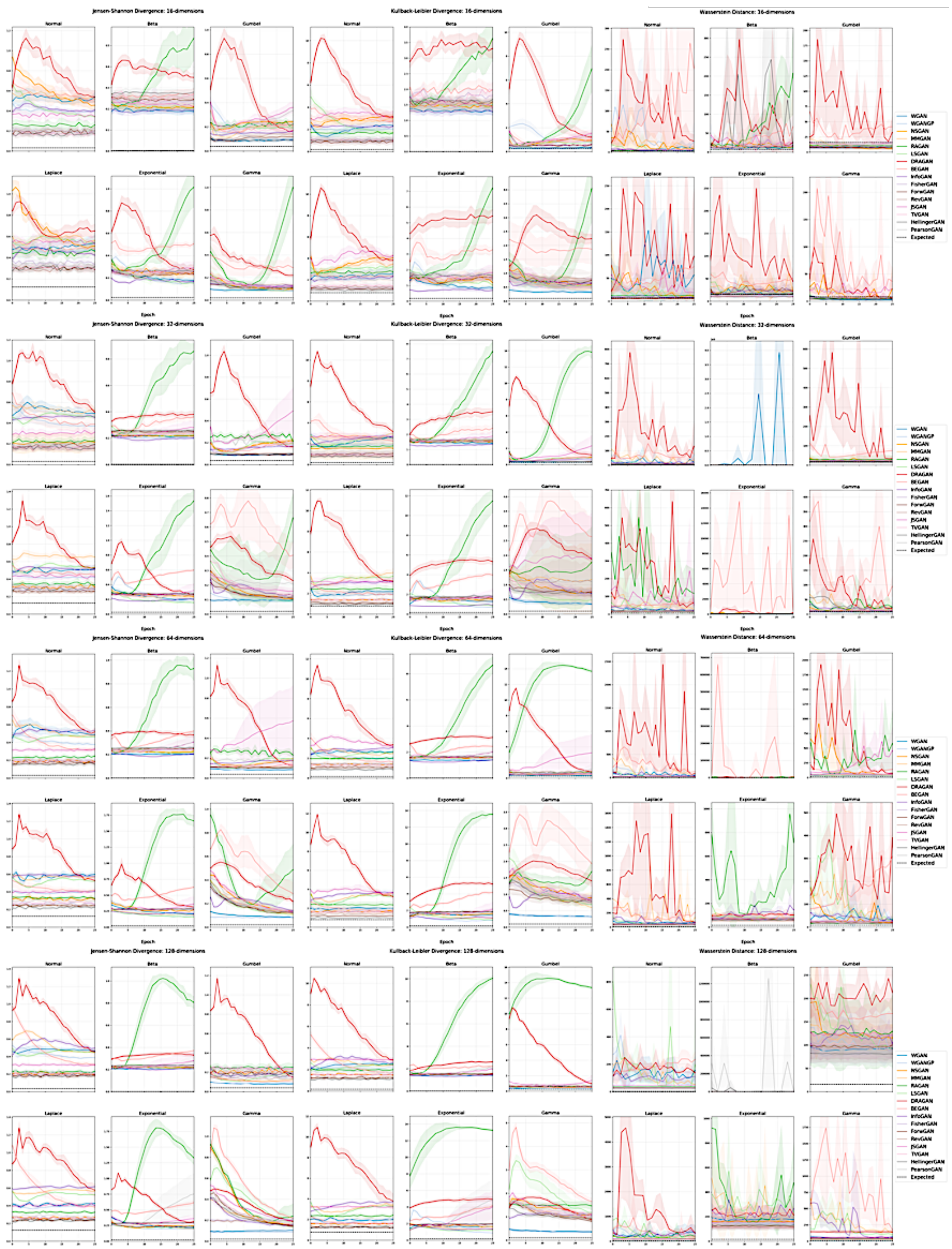

Figure 1: Performance of GAN variants for their best hyperparameter settings, respectively, trained on 100,000 samples across normal, beta, Gumbel, Laplace, exponential, and gamma multivariate distributions of dimensionalities $N = 16$ (rows 1 and 2), $N = 32$ (rows 3 and 4), $N = 64$ (rows 5 and 6), and $N = 128$ (rows 7 and 8). Plots display metric performance as a function of epoch. Shaded areas represent the region of the 95% confidence interval of the respective model computed over 20 trials. "Expected" indicates the empirical average divergence of a generated batch of size $b = 1024$ for the given distribution across 20 trials where samples are generated independently, i.e. one at a time. Best viewed in color.

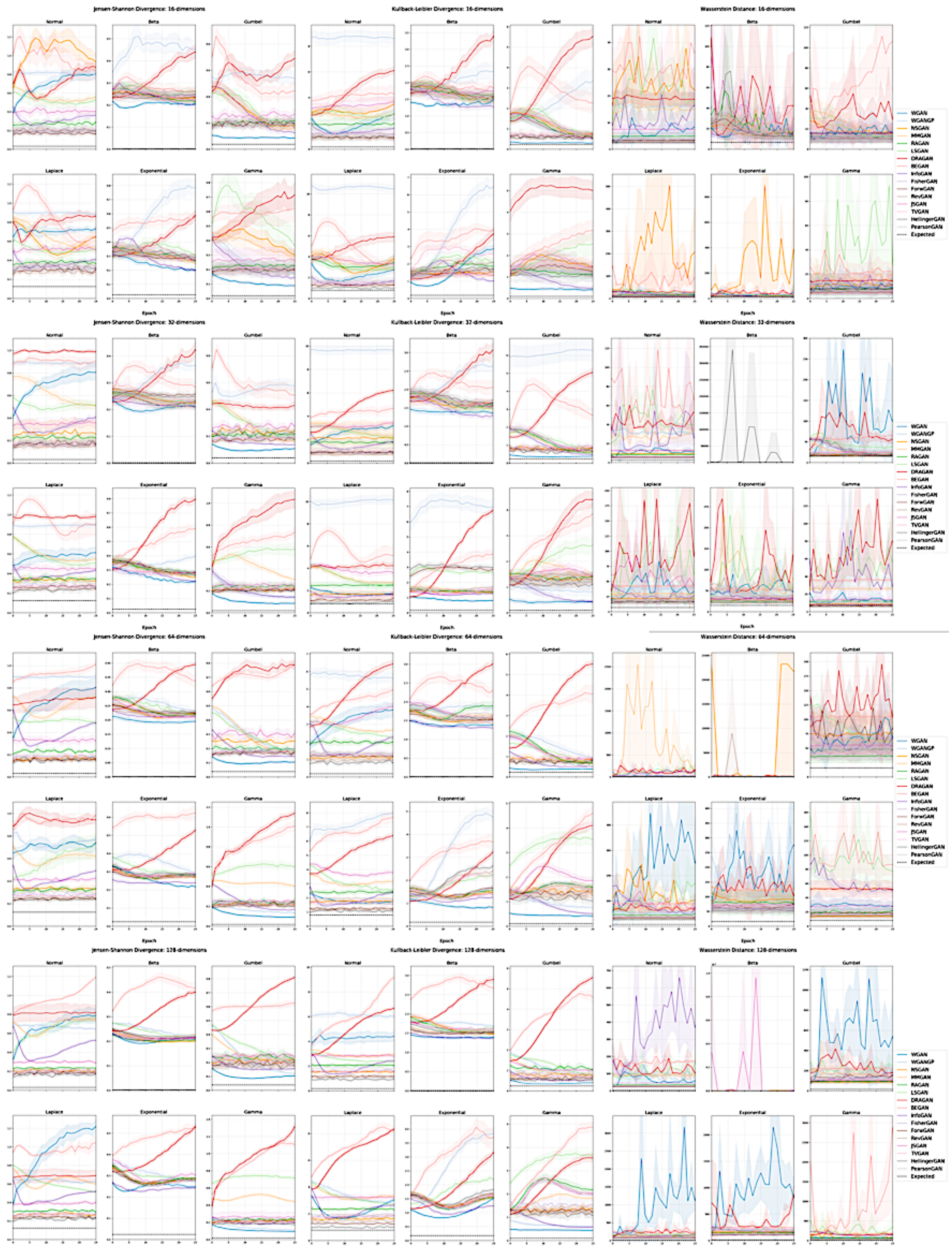

Figure 2: Performance of GAN variants for their best hyperparameter settings, respectively, trained on 10,000 samples across normal, beta, Gumbel, Laplace, exponential, and gamma multivariate distributions of dimensionalities $N = 16$ (rows 1 and 2), $N = 32$ (rows 3 and 4), $N = 64$ (rows 5 and 6), and $N = 128$ (rows 7 and 8). Plots display metric performance as a function of epoch. Shaded areas represent the region of the 95% confidence interval of the respective model computed over 20 trials. "Expected" indicates the empirical average divergence of a generated batch of size $b = 1024$ for the given distribution across 20 trials where samples are generated independently, i.e. one at a time. Best viewed in color.

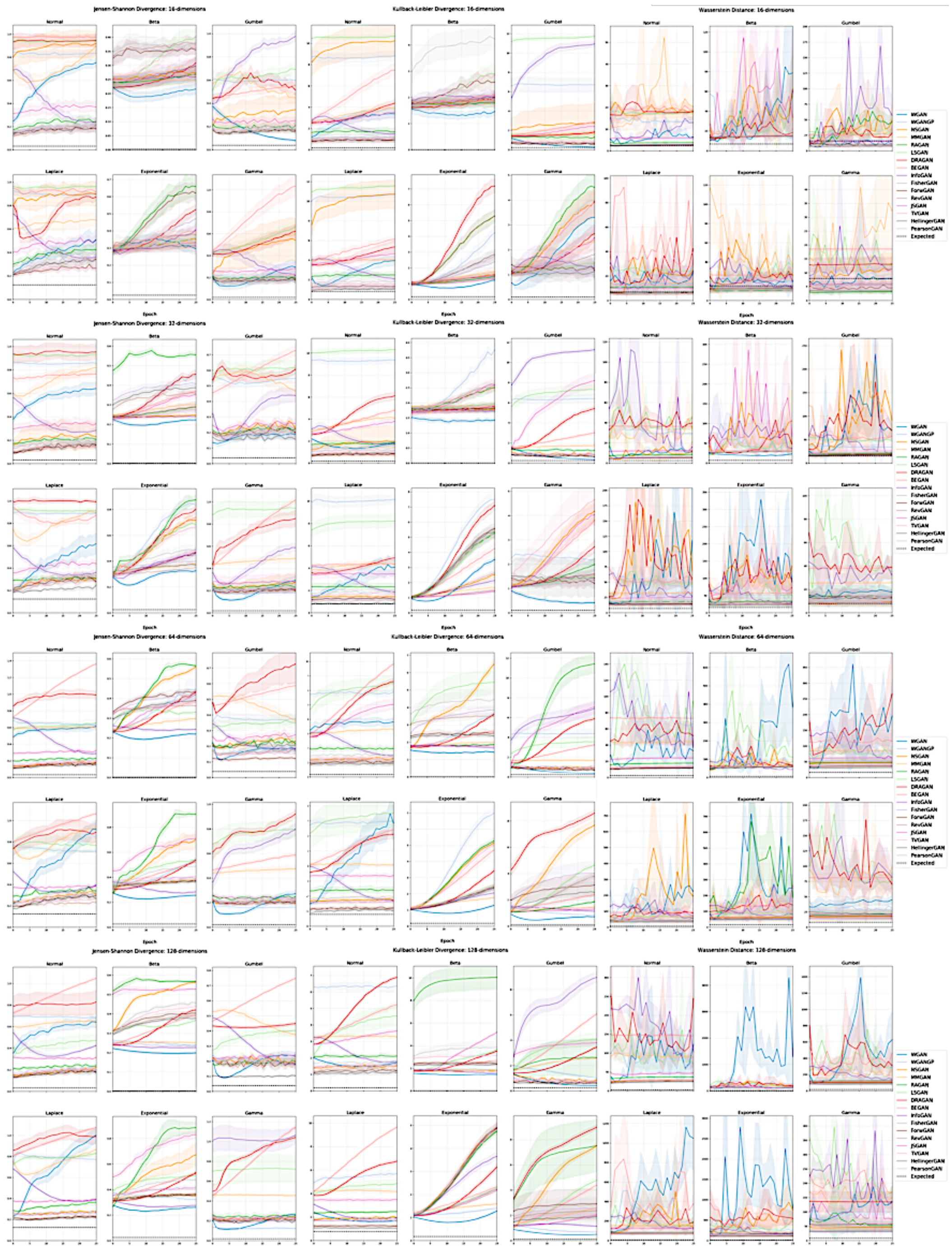

Figure 3: Performance of GAN variants for their best hyperparameter settings, respectively, trained on 1,000 samples across normal, beta, Gumbel, Laplace, exponential, and gamma multivariate distributions of dimensionalities $N = 16$ (rows 1 and 2), $N = 32$ (rows 3 and 4), $N = 64$ (rows 5 and 6), and $N = 128$ (rows 7 and 8). Plots display metric performance as a function of epoch. Shaded areas represent the region of the 95% confidence interval of the respective model computed over 20 trials. “Expected” indicates the empirical average divergence of a generated batch of size $b = 1024$ for the given distribution across 20 trials where samples are generated independently, i.e. one at a time. Best viewed in color.

B Number of training samples can have a large impact on performance

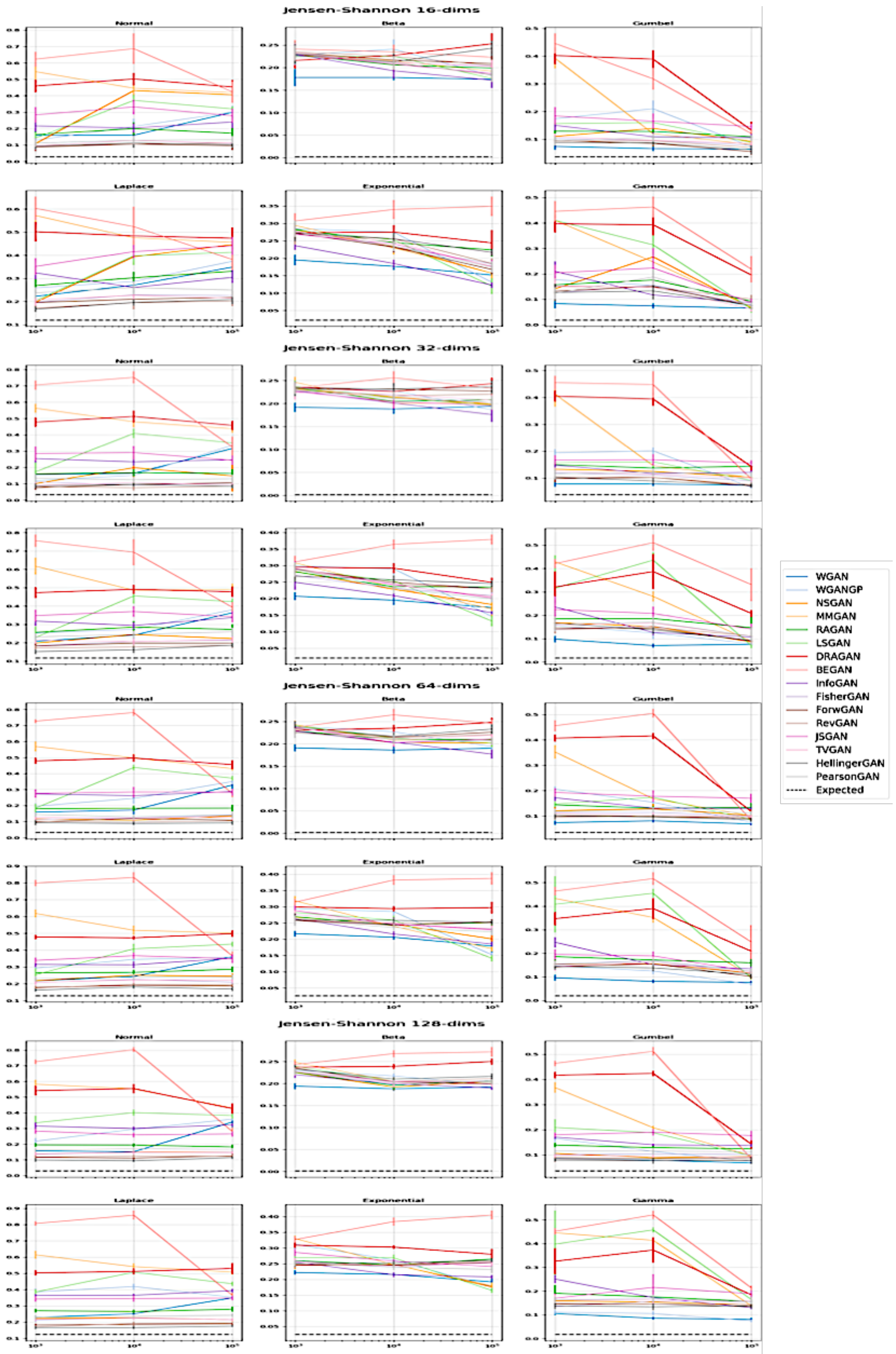

Figure 4: Confidence intervals of the Jensen-Shannon divergence performances for the best performing hyperparameter over 20 trials as a function of sample size for dimensionalities $N = 16$ (rows 1 and 2), $N = 32$ (rows 3 and 4), $N = 64$ (rows 5 and 6), and $N = 128$ (rows 7 and 8). “Expected” indicates the empirical average divergence of a generated batch of size $b = 1024$ for the given distribution would be across 20 trials where samples are generated independently, i.e. one at a time. Best viewed in color.

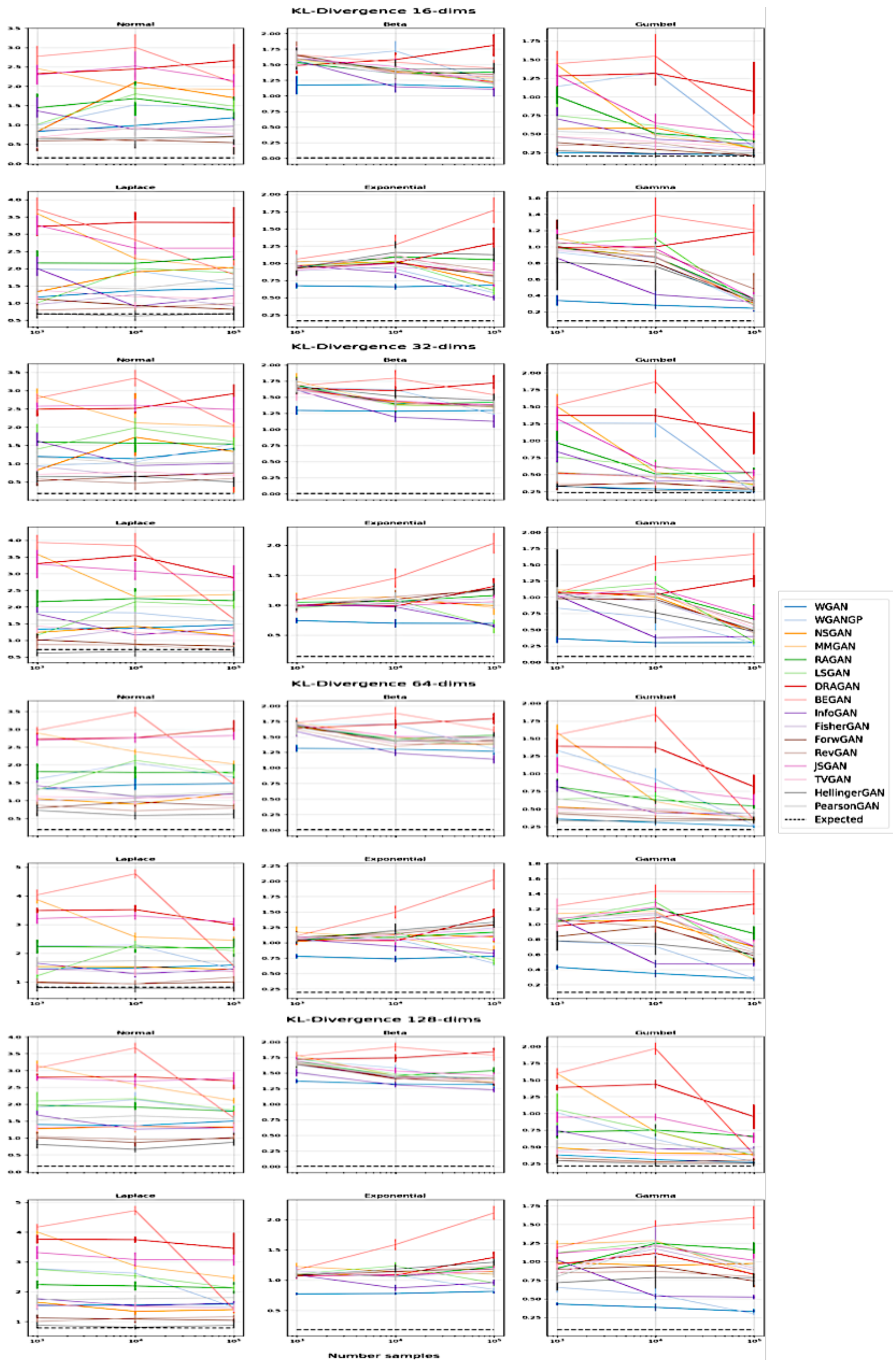

Figure 5: Confidence intervals of the Kullback-Leibler divergence performances for the best performing hyperparameter over 20 trials as a function of sample size for dimensionalities $N = 16$ (rows 1 and 2), $N = 32$ (rows 3 and 4), $N = 64$ (rows 5 and 6), and $N = 128$ (rows 7 and 8). "Expected" indicates the empirical average divergence of a generated batch of size $b = 1024$ for the given distribution would be across 20 trials where samples are generated independently, i.e. one at a time. Best viewed in color.

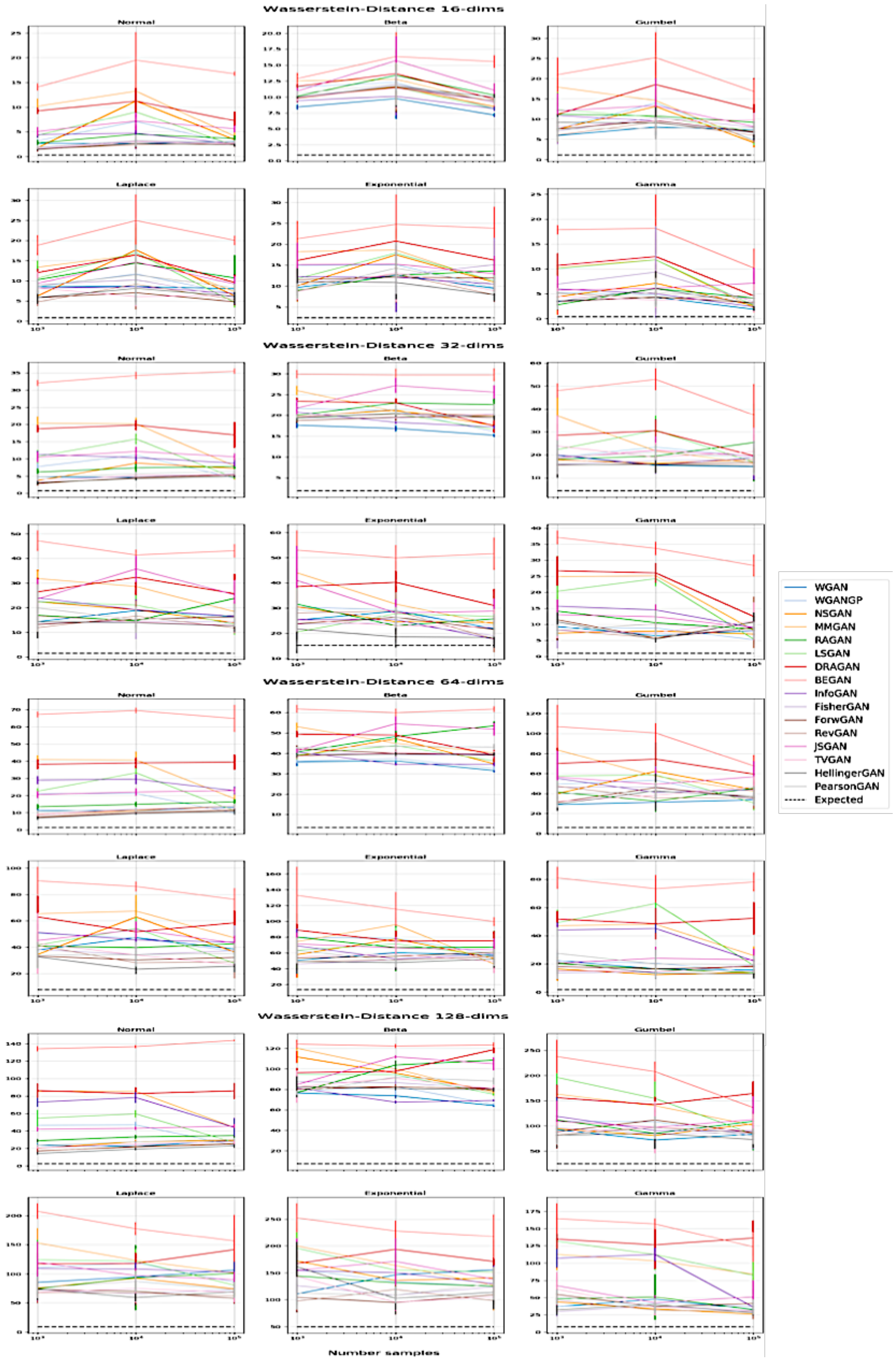

Figure 6: Confidence intervals of the Wasserstein Distance performances for the best performing hyperparameter over 20 trials as a function of sample size for dimensionalities $N = 16$ (rows 1 and 2), $N = 32$ (rows 3 and 4), $N = 64$ (rows 5 and 6), and $N = 128$ (rows 7 and 8). “Expected” indicates the empirical average divergence of a generated batch of size $b = 1024$ for the given distribution would be across 20 trials where samples are generated independently, i.e. one at a time. Best viewed in color.

C Evaluation and comparisons are metric-dependent

Model	Normal	Beta	Gumbel	Laplace	Exponential	Gamma
WGAN	[7, 7, 13] [8, 9, 10] [7, 2, 11]	[0, 0, 5] [0, 1, 4] [0, 0, 4]	[0, 0, 1] [9, 5, 7] [1, 2, 9]	[5, 7, 13] [9, 9, 9] [8, 8, 11]	[0, 0, 3] [1, 3, 4] [3, 6, 4]	[0, 0, 2] [0, 0, 1] [5, 0, 2]
WGANGP	[8, 10, 14] [9, 10, 11] [9, 9, 0]	[13, 2, 0] [11, 3, 1] [2, 1, 1]	[13, 8, 4] [13, 12, 9] [8, 11, 3]	[9, 9, 14] [12, 11, 10] [10, 10, 2]	[13, 1, 0] [12, 4, 1] [9, 4, 3]	[9, 1, 0] [9, 4, 0] [10, 3, 0]
NSGAN	[4, 6, 6] [5, 1, 5] [4, 7, 10]	[6, 3, 4] [7, 2, 3] [10, 4, 6]	[2, 4, 6] [1, 4, 4] [5, 6, 1]	[2, 8, 5] [5, 4, 5] [4, 9, 3]	[5, 4, 6] [2, 0, 5] [4, 5, 10]	[6, 2, 3] [7, 8, 5] [6, 4, 4]
MMGAN	[14, 14, 11] [14, 13, 12] [14, 13, 12]	[7, 5, 6] [10, 6, 5] [14, 5, 5]	[12, 11, 7] [12, 9, 10] [14, 4, 2]	[14, 13, 11] [13, 13, 13] [14, 11, 5]	[3, 5, 4] [14, 11, 6] [14, 13, 12]	[13, 5, 4] [14, 9, 2] [13, 12, 1]
RAGAN	[10, 8, 7] [6, 7, 7] [8, 10, 13]	[11, 13, 14] [12, 13, 14] [8, 15, 13]	[8, 9, 13] [7, 8, 8] [4, 12, 10]	[10, 10, 9] [6, 8, 7] [7, 4, 13]	[6, 13, 14] [11, 14, 14] [2, 8, 13]	[10, 9, 14] [10, 6, 11] [0, 6, 12]
LSGAN	[9, 12, 8] [10, 12, 9] [11, 8, 1]	[9, 4, 2] [9, 4, 0] [11, 3, 3]	[9, 10, 8] [11, 11, 11] [11, 8, 0]	[6, 12, 8] [8, 12, 11] [11, 14, 0]	[2, 2, 1] [13, 10, 3] [11, 12, 7]	[14, 7, 1] [12, 11, 3] [12, 13, 5]
DRAGAN	[13, 13, 10] [13, 14, 13] [13, 14, 7]	[14, 14, 1] [14, 14, 10] [5, 7, 0]	[14, 15, 11] [14, 15, 15] [13, 14, 4]	[13, 15, 10] [14, 15, 15] [13, 13, 10]	[14, 14, 2] [10, 13, 10] [13, 14, 0]	[12, 14, 11] [11, 14, 14] [14, 14, 6]
BEGAN	[15, 15, 15] [15, 15, 15] [15, 15, 15]	[15, 15, 15] [15, 15, 15] [15, 14, 14]	[15, 14, 15] [15, 14, 14] [15, 15, 15]	[15, 14, 15] [15, 14, 12] [15, 15, 15]	[15, 15, 15] [15, 15, 15] [15, 15, 15]	[15, 15, 15] [15, 15, 15] [15, 15, 15]
InfoGAN	[11, 9, 9] [11, 8, 8] [10, 11, 4]	[1, 1, 3] [1, 0, 2] [1, 2, 2]	[10, 12, 14] [10, 10, 13] [9, 7, 7]	[11, 6, 6] [10, 5, 8] [9, 5, 8]	[1, 3, 5] [6, 1, 0] [7, 7, 8]	[4, 3, 5] [8, 1, 7] [11, 10, 3]
FisherGAN	[5, 2, 5] [2, 4, 4] [5, 5, 9]	[4, 8, 8] [5, 10, 7] [7, 12, 7]	[5, 5, 5] [2, 6, 5] [0, 0, 11]	[3, 4, 4] [2, 6, 1] [6, 7, 6]	[10, 8, 11] [4, 2, 8] [8, 9, 11]	[5, 10, 7] [3, 5, 10] [8, 5, 9]
ForwGAN	[0, 3, 0] [0, 0, 6] [0, 1, 3]	[5, 9, 10] [3, 8, 11] [3, 8, 10]	[3, 1, 0] [4, 7, 1] [7, 10, 8]	[4, 3, 2] [3, 1, 3] [2, 2, 1]	[12, 6, 7] [8, 9, 9] [10, 3, 5]	[1, 6, 8] [4, 7, 8] [7, 8, 8]
RevGAN	[2, 4, 2] [4, 3, 1] [2, 0, 8]	[2, 10, 9] [4, 7, 6] [6, 9, 11]	[4, 6, 2] [5, 3, 2] [2, 9, 5]	[1, 0, 1] [1, 2, 2] [0, 1, 4]	[4, 9, 12] [7, 5, 13] [5, 1, 1]	[2, 4, 12] [2, 2, 13] [2, 1, 13]
JSGAN	[12, 11, 12] [12, 11, 14] [12, 12, 14]	[12, 11, 11] [13, 12, 13] [13, 10, 15]	[11, 13, 12] [8, 13, 12] [12, 13, 13]	[12, 11, 12] [11, 10, 14] [12, 12, 14]	[11, 10, 8] [9, 12, 11] [12, 11, 14]	[11, 13, 13] [13, 12, 12] [9, 11, 14]
TVGAN	[3, 0, 3] [3, 5, 3] [3, 4, 5]	[8, 6, 12] [6, 9, 9] [9, 6, 8]	[6, 2, 9] [6, 2, 6] [10, 5, 14]	[7, 2, 3] [7, 7, 4] [5, 0, 9]	[7, 7, 9] [3, 7, 7] [0, 0, 6]	[7, 11, 6] [6, 10, 4] [1, 2, 11]
HellingerGAN	[1, 1, 1] [1, 2, 0] [1, 3, 2]	[3, 7, 7] [2, 5, 8] [4, 11, 12]	[1, 3, 3] [0, 1, 0] [3, 3, 6]	[0, 1, 0] [0, 0, 0] [3, 3, 7]	[8, 12, 13] [5, 8, 12] [6, 2, 2]	[3, 8, 9] [1, 3, 9] [4, 7, 7]
PearsonGAN	[6, 5, 4] [7, 6, 2] [6, 6, 6]	[10, 12, 13] [8, 11, 12] [12, 13, 9]	[7, 7, 10] [3, 0, 3] [6, 1, 12]	[8, 5, 7] [4, 3, 6] [1, 6, 12]	[9, 11, 10] [0, 6, 2] [1, 10, 9]	[8, 12, 10] [5, 13, 6] [3, 9, 10]

Table 2: Performance-based relative ranking of GAN variants for 1,000 samples (first entry of each row), 10,000 samples (second entry), and 100,000 samples (third entry) with respect to Jensen-Shannon divergence (first row), Kullback-Leibler divergence (second row), and Wasserstein distance (third row) for the best hyperparameter settings for dimensionality $N = 128$. Note that differences between performances are sometimes marginal. Dimensionalities $N \in \{16, 32, 64\}$ yielded similar ranking results to these with slight variations.

Model	Normal	Beta	Gumbel	Laplace	Exponential	Gamma
WGAN	0.128 ± 0.019	0.099 ± 0.010	0.062 ± 0.007	0.201 ± 0.019	0.156 ± 0.010	0.078 ± 0.011
	0.160 ± 0.029	0.109 ± 0.011	0.066 ± 0.009	0.271 ± 0.027	0.128 ± 0.017	0.059 ± 0.009
	0.299 ± 0.037	0.114 ± 0.018	0.064 ± 0.014	0.349 ± 0.050	0.108 ± 0.013	0.062 ± 0.013
WGANGP	0.129 ± 0.026	0.190 ± 0.020	0.174 ± 0.023	0.259 ± 0.042	0.248 ± 0.019	0.143 ± 0.032
	0.213 ± 0.048	0.121 ± 0.011	0.092 ± 0.013	0.287 ± 0.033	0.122 ± 0.014	0.068 ± 0.012
	0.291 ± 0.022	0.066 ± 0.016	0.073 ± 0.011	0.341 ± 0.012	0.083 ± 0.010	0.051 ± 0.011
NSGAN	0.106 ± 0.021	0.161 ± 0.019	0.079 ± 0.009	0.197 ± 0.022	0.213 ± 0.013	0.125 ± 0.025
	0.120 ± 0.018	0.120 ± 0.013	0.085 ± 0.012	0.277 ± 0.028	0.146 ± 0.015	0.089 ± 0.020
	0.126 ± 0.025	0.105 ± 0.013	0.080 ± 0.013	0.242 ± 0.021	0.110 ± 0.014	0.060 ± 0.016
MMGAN	0.422 ± 0.040	0.168 ± 0.014	0.148 ± 0.026	0.444 ± 0.033	0.202 ± 0.013	0.311 ± 0.037
	0.401 ± 0.025	0.142 ± 0.032	0.100 ± 0.009	0.440 ± 0.036	0.142 ± 0.018	0.098 ± 0.023
	0.246 ± 0.033	0.110 ± 0.027	0.081 ± 0.009	0.334 ± 0.042	0.107 ± 0.015	0.060 ± 0.009
RaGAN	0.160 ± 0.029	0.180 ± 0.019	0.100 ± 0.013	0.269 ± 0.027	0.216 ± 0.023	0.146 ± 0.035
	0.169 ± 0.027	0.184 ± 0.016	0.099 ± 0.021	0.284 ± 0.023	0.220 ± 0.015	0.128 ± 0.028
	0.156 ± 0.029	0.199 ± 0.016	0.104 ± 0.014	0.293 ± 0.029	0.224 ± 0.019	0.092 ± 0.020
LSGAN	0.142 ± 0.035	0.170 ± 0.025	0.115 ± 0.015	0.204 ± 0.017	0.198 ± 0.019	0.340 ± 0.052
	0.306 ± 0.021	0.127 ± 0.014	0.094 ± 0.019	0.333 ± 0.022	0.128 ± 0.020	0.102 ± 0.027
	0.206 ± 0.016	0.080 ± 0.026	0.088 ± 0.010	0.288 ± 0.019	0.090 ± 0.012	0.056 ± 0.014
DRAGAN	0.354 ± 0.041	0.212 ± 0.025	0.272 ± 0.031	0.392 ± 0.032	0.260 ± 0.021	0.301 ± 0.045
	0.385 ± 0.064	0.198 ± 0.026	0.248 ± 0.051	0.457 ± 0.037	0.259 ± 0.054	0.216 ± 0.035
	0.239 ± 0.017	0.069 ± 0.020	0.098 ± 0.010	0.297 ± 0.021	0.096 ± 0.020	0.087 ± 0.016
BEGAN	0.549 ± 0.050	0.231 ± 0.018	0.329 ± 0.069	0.507 ± 0.051	0.301 ± 0.026	0.446 ± 0.040
	0.457 ± 0.056	0.226 ± 0.016	0.179 ± 0.057	0.445 ± 0.050	0.340 ± 0.027	0.287 ± 0.061
	0.348 ± 0.055	0.223 ± 0.017	0.120 ± 0.025	0.381 ± 0.041	0.307 ± 0.025	0.218 ± 0.052
InfoGAN	0.190 ± 0.023	0.122 ± 0.019	0.113 ± 0.012	0.261 ± 0.024	0.151 ± 0.011	0.119 ± 0.019
	0.204 ± 0.028	0.105 ± 0.011	0.104 ± 0.007	0.261 ± 0.026	0.134 ± 0.014	0.089 ± 0.019
	0.202 ± 0.032	0.085 ± 0.015	0.104 ± 0.005	0.243 ± 0.022	0.102 ± 0.010	0.066 ± 0.018
FisherGAN	0.112 ± 0.030	0.156 ± 0.013	0.083 ± 0.013	0.197 ± 0.029	0.223 ± 0.020	0.117 ± 0.023
	0.109 ± 0.027	0.160 ± 0.011	0.081 ± 0.010	0.221 ± 0.026	0.193 ± 0.013	0.124 ± 0.024
	0.110 ± 0.022	0.176 ± 0.021	0.078 ± 0.012	0.222 ± 0.020	0.184 ± 0.009	0.078 ± 0.015
ForwGAN	0.083 ± 0.016	0.159 ± 0.017	0.073 ± 0.008	0.196 ± 0.021	0.233 ± 0.020	0.107 ± 0.019
	0.106 ± 0.021	0.179 ± 0.020	0.075 ± 0.013	0.210 ± 0.021	0.187 ± 0.011	0.098 ± 0.022
	0.096 ± 0.026	0.181 ± 0.030	0.055 ± 0.011	0.205 ± 0.021	0.168 ± 0.014	0.075 ± 0.014
RevGAN	0.097 ± 0.027	0.147 ± 0.022	0.074 ± 0.010	0.172 ± 0.014	0.206 ± 0.022	0.104 ± 0.021
	0.103 ± 0.025	0.173 ± 0.021	0.080 ± 0.009	0.188 ± 0.018	0.197 ± 0.013	0.087 ± 0.019
	0.101 ± 0.015	0.174 ± 0.016	0.062 ± 0.011	0.187 ± 0.019	0.185 ± 0.015	0.085 ± 0.019
JSGAN	0.245 ± 0.034	0.183 ± 0.016	0.116 ± 0.020	0.352 ± 0.036	0.229 ± 0.020	0.147 ± 0.026
	0.243 ± 0.035	0.175 ± 0.013	0.124 ± 0.019	0.324 ± 0.025	0.195 ± 0.046	0.150 ± 0.034
	0.257 ± 0.038	0.187 ± 0.012	0.097 ± 0.018	0.339 ± 0.055	0.167 ± 0.026	0.086 ± 0.016
TVGAN	0.095 ± 0.022	0.165 ± 0.018	0.081 ± 0.014	0.204 ± 0.024	0.215 ± 0.022	0.126 ± 0.026
	0.098 ± 0.016	0.157 ± 0.012	0.079 ± 0.009	0.205 ± 0.025	0.188 ± 0.013	0.123 ± 0.024
	0.101 ± 0.016	0.183 ± 0.016	0.085 ± 0.013	0.204 ± 0.035	0.176 ± 0.012	0.062 ± 0.011
HellingerGAN	0.088 ± 0.021	0.148 ± 0.014	0.067 ± 0.011	0.167 ± 0.012	0.215 ± 0.024	0.102 ± 0.022
	0.097 ± 0.018	0.152 ± 0.013	0.072 ± 0.015	0.196 ± 0.014	0.210 ± 0.012	0.104 ± 0.018
	0.099 ± 0.022	0.159 ± 0.018	0.062 ± 0.016	0.179 ± 0.021	0.212 ± 0.022	0.076 ± 0.014
PearsonGAN	0.112 ± 0.023	0.172 ± 0.021	0.087 ± 0.013	0.238 ± 0.024	0.217 ± 0.024	0.124 ± 0.023
	0.118 ± 0.021	0.177 ± 0.014	0.088 ± 0.008	0.256 ± 0.027	0.203 ± 0.018	0.121 ± 0.026
	0.100 ± 0.013	0.180 ± 0.020	0.084 ± 0.009	0.250 ± 0.030	0.175 ± 0.016	0.074 ± 0.015

Table 3: Confidence intervals of the best average minimum Jensen-Shannon divergence across 20 trials for 1,000 samples (first row), 10,000 samples (second row), and 100,000 samples (third row) of the best hyperparameter settings for dimensionality $N = 128$.

Model	Normal	Beta	Gumbel	Laplace	Exponential	Gamma
WGAN	0.832 ± 0.295	0.445 ± 0.043	0.251 ± 0.037	1.174 ± 0.249	0.609 ± 0.061	0.312 ± 0.038
	0.985 ± 0.188	0.521 ± 0.082	0.232 ± 0.025	1.359 ± 0.221	0.639 ± 0.052	0.229 ± 0.037
	1.185 ± 0.194	0.566 ± 0.083	0.216 ± 0.025	1.430 ± 0.285	0.552 ± 0.040	0.244 ± 0.040
WGANGP	0.989 ± 0.280	0.985 ± 0.119	1.139 ± 0.303	1.916 ± 0.351	0.955 ± 0.122	0.904 ± 0.144
	1.520 ± 0.430	0.610 ± 0.091	0.440 ± 0.085	1.943 ± 0.412	0.661 ± 0.142	0.393 ± 0.177
	1.380 ± 0.134	0.401 ± 0.131	0.318 ± 0.041	1.533 ± 0.184	0.463 ± 0.042	0.216 ± 0.042
NSGAN	0.544 ± 0.238	0.814 ± 0.256	0.177 ± 0.010	0.891 ± 0.202	0.687 ± 0.206	0.603 ± 0.299
	0.381 ± 0.181	0.583 ± 0.080	0.221 ± 0.075	0.874 ± 0.256	0.546 ± 0.170	0.534 ± 0.204
	0.522 ± 0.179	0.473 ± 0.064	0.165 ± 0.011	0.788 ± 0.249	0.556 ± 0.078	0.302 ± 0.069
MMGAN	2.458 ± 0.184	0.932 ± 0.114	0.864 ± 0.228	2.796 ± 0.373	1.028 ± 0.115	1.108 ± 0.158
	1.946 ± 0.219	0.697 ± 0.094	0.374 ± 0.041	2.290 ± 0.347	0.999 ± 0.146	0.584 ± 0.185
	1.449 ± 0.327	0.572 ± 0.127	0.310 ± 0.076	2.013 ± 0.155	0.598 ± 0.086	0.254 ± 0.038
RaGAN	0.730 ± 0.299	0.992 ± 0.126	0.211 ± 0.077	0.900 ± 0.187	0.940 ± 0.101	0.935 ± 0.368
	0.795 ± 0.342	1.067 ± 0.129	0.262 ± 0.186	1.167 ± 0.323	1.034 ± 0.126	0.451 ± 0.160
	0.741 ± 0.242	1.183 ± 0.337	0.221 ± 0.080	1.040 ± 0.243	1.052 ± 0.126	0.357 ± 0.085
LSGAN	1.012 ± 0.369	0.901 ± 0.155	0.616 ± 0.157	1.006 ± 0.196	0.950 ± 0.062	1.035 ± 0.118
	1.608 ± 0.316	0.633 ± 0.103	0.390 ± 0.112	2.004 ± 0.324	0.925 ± 0.221	0.675 ± 0.257
	1.073 ± 0.102	0.365 ± 0.128	0.316 ± 0.038	1.728 ± 0.221	0.495 ± 0.070	0.274 ± 0.075
DRAGAN	2.326 ± 0.230	1.075 ± 0.153	1.281 ± 0.120	3.223 ± 0.270	0.930 ± 0.089	0.993 ± 0.094
	2.444 ± 0.213	1.367 ± 0.187	1.314 ± 0.162	3.348 ± 0.301	1.008 ± 0.144	1.001 ± 0.092
	1.608 ± 0.226	0.683 ± 0.192	0.594 ± 0.139	2.119 ± 0.376	0.732 ± 0.286	0.879 ± 0.326
BEGAN	2.774 ± 0.268	1.360 ± 0.118	1.445 ± 0.170	3.721 ± 0.347	1.060 ± 0.129	1.144 ± 0.131
	3.004 ± 0.339	1.394 ± 0.147	1.008 ± 0.452	2.389 ± 0.421	1.269 ± 0.149	1.391 ± 0.217
	2.099 ± 0.368	1.446 ± 0.134	0.589 ± 0.183	1.834 ± 0.421	1.645 ± 0.257	1.209 ± 0.316
InfoGAN	1.021 ± 0.187	0.510 ± 0.103	0.561 ± 0.121	1.464 ± 0.220	0.826 ± 0.118	0.664 ± 0.208
	0.882 ± 0.158	0.505 ± 0.115	0.371 ± 0.024	0.906 ± 0.110	0.593 ± 0.081	0.336 ± 0.051
	0.965 ± 0.197	0.428 ± 0.109	0.370 ± 0.022	1.213 ± 0.076	0.456 ± 0.051	0.321 ± 0.044
FisherGAN	0.391 ± 0.173	0.709 ± 0.060	0.171 ± 0.015	0.779 ± 0.240	0.803 ± 0.290	0.381 ± 0.149
	0.609 ± 0.181	0.818 ± 0.061	0.235 ± 0.081	0.906 ± 0.182	0.621 ± 0.202	0.394 ± 0.180
	0.483 ± 0.172	0.646 ± 0.166	0.176 ± 0.013	0.637 ± 0.129	0.716 ± 0.224	0.346 ± 0.100
ForwGAN	0.321 ± 0.137	0.680 ± 0.202	0.190 ± 0.009	0.771 ± 0.219	0.909 ± 0.218	0.397 ± 0.134
	0.377 ± 0.138	0.808 ± 0.160	0.232 ± 0.151	0.653 ± 0.150	0.873 ± 0.210	0.466 ± 0.280
	0.541 ± 0.310	0.766 ± 0.219	0.159 ± 0.014	0.735 ± 0.265	0.712 ± 0.150	0.334 ± 0.087
RevGAN	0.518 ± 0.174	0.690 ± 0.133	0.191 ± 0.074	0.698 ± 0.195	0.820 ± 0.188	0.352 ± 0.132
	0.519 ± 0.211	0.745 ± 0.197	0.214 ± 0.019	0.741 ± 0.208	0.779 ± 0.222	0.369 ± 0.151
	0.370 ± 0.170	0.630 ± 0.175	0.150 ± 0.013	0.691 ± 0.135	0.863 ± 0.210	0.400 ± 0.137
JSGAN	1.919 ± 0.639	1.014 ± 0.115	0.219 ± 0.071	1.705 ± 0.445	0.905 ± 0.087	1.050 ± 0.137
	1.560 ± 0.439	0.904 ± 0.175	0.521 ± 0.217	1.855 ± 0.491	0.998 ± 0.130	0.687 ± 0.136
	1.879 ± 0.384	1.034 ± 0.226	0.346 ± 0.130	2.077 ± 0.561	0.858 ± 0.201	0.359 ± 0.083
TVGAN	0.486 ± 0.188	0.762 ± 0.244	0.201 ± 0.075	0.911 ± 0.273	0.694 ± 0.242	0.590 ± 0.237
	0.602 ± 0.187	0.800 ± 0.248	0.197 ± 0.079	0.962 ± 0.240	0.798 ± 0.214	0.581 ± 0.220
	0.460 ± 0.189	0.670 ± 0.181	0.180 ± 0.015	0.752 ± 0.169	0.594 ± 0.200	0.273 ± 0.058
HellingerGAN	0.377 ± 0.208	0.548 ± 0.112	0.155 ± 0.012	0.645 ± 0.195	0.801 ± 0.170	0.337 ± 0.103
	0.409 ± 0.190	0.678 ± 0.195	0.187 ± 0.026	0.628 ± 0.148	0.804 ± 0.238	0.379 ± 0.111
	0.352 ± 0.100	0.659 ± 0.200	0.149 ± 0.010	0.586 ± 0.191	0.855 ± 0.152	0.337 ± 0.076
PearsonGAN	0.764 ± 0.242	0.869 ± 0.216	0.177 ± 0.015	0.874 ± 0.286	0.578 ± 0.147	0.409 ± 0.202
	0.600 ± 0.229	0.815 ± 0.202	0.175 ± 0.025	0.862 ± 0.415	0.786 ± 0.214	0.744 ± 0.169
	0.391 ± 0.143	0.867 ± 0.262	0.154 ± 0.017	0.820 ± 0.246	0.481 ± 0.097	0.318 ± 0.099

Table 4: Confidence intervals of the best average minimum Kullback-Leibler divergence across 20 trials for 1,000 samples (first row), 10,000 samples (second row), and 100,000 samples (third row) of the best hyperparameter settings for dimensionality $N = 128$.

Model	Normal	Beta	Gumbel	Laplace	Exponential	Gamma
WGAN	2.376 ± 0.428	6.301 ± 0.543	5.910 ± 1.368	7.698 ± 2.711	9.615 ± 2.024	3.235 ± 1.053
	2.655 ± 0.950	6.777 ± 2.090	8.019 ± 1.944	8.577 ± 3.843	11.899 ± 5.086	2.995 ± 1.149
	3.042 ± 0.449	5.927 ± 0.366	7.037 ± 2.640	6.988 ± 1.975	8.405 ± 1.402	1.881 ± 0.543
WGANGP	3.191 ± 0.400	8.336 ± 0.654	7.611 ± 2.508	8.480 ± 3.452	11.127 ± 2.078	4.279 ± 1.734
	4.195 ± 1.431	7.808 ± 2.472	9.871 ± 4.512	9.882 ± 5.517	11.367 ± 6.599	3.672 ± 1.154
	2.043 ± 0.147	5.038 ± 0.290	4.768 ± 1.626	4.941 ± 1.163	8.142 ± 3.307	1.348 ± 0.396
NSGAN	1.749 ± 0.413	9.099 ± 0.734	7.445 ± 2.463	6.187 ± 1.414	10.076 ± 3.777	3.401 ± 1.031
	3.358 ± 1.237	8.643 ± 2.142	8.464 ± 2.808	8.950 ± 3.988	11.447 ± 4.605	3.737 ± 1.477
	3.029 ± 0.229	7.469 ± 0.535	4.117 ± 1.023	5.327 ± 1.703	10.218 ± 3.821	2.223 ± 0.816
MMGAN	10.201 ± 1.561	10.464 ± 0.755	11.390 ± 2.192	13.033 ± 1.864	17.059 ± 3.348	10.312 ± 1.613
	5.495 ± 1.363	10.551 ± 3.462	8.330 ± 3.214	10.662 ± 5.150	18.233 ± 6.909	5.926 ± 5.555
	3.110 ± 0.452	6.617 ± 0.661	4.452 ± 1.029	5.534 ± 1.823	10.873 ± 2.759	1.843 ± 0.506
RaGAN	2.740 ± 0.566	8.983 ± 0.788	7.380 ± 2.084	7.509 ± 2.185	8.976 ± 2.146	2.791 ± 1.016
	4.519 ± 1.452	13.032 ± 4.097	10.823 ± 4.832	8.052 ± 2.342	12.526 ± 3.182	4.056 ± 1.408
	3.537 ± 0.381	10.291 ± 0.733	7.178 ± 1.257	7.852 ± 2.715	10.936 ± 2.011	3.654 ± 1.024
LSGAN	4.280 ± 0.518	9.232 ± 0.563	10.503 ± 3.403	8.753 ± 2.167	11.754 ± 3.206	9.023 ± 0.929
	4.096 ± 1.636	8.427 ± 2.414	8.978 ± 4.606	13.346 ± 4.851	15.695 ± 5.691	7.326 ± 5.143
	2.286 ± 0.190	5.827 ± 0.778	3.460 ± 0.814	4.262 ± 0.798	9.539 ± 3.441	2.345 ± 0.962
DRAGAN	7.051 ± 0.851	8.892 ± 0.572	11.208 ± 1.052	11.004 ± 1.971	15.461 ± 2.439	10.695 ± 2.428
	9.638 ± 2.113	11.497 ± 4.364	17.183 ± 6.580	12.998 ± 3.925	19.594 ± 9.288	9.541 ± 5.206
	2.704 ± 0.338	4.614 ± 0.641	5.762 ± 2.537	6.923 ± 1.181	7.696 ± 2.245	2.713 ± 0.689
BEGAN	14.073 ± 0.759	11.215 ± 0.477	20.592 ± 2.197	18.819 ± 2.514	21.311 ± 4.300	16.569 ± 0.759
	19.480 ± 5.667	12.881 ± 3.939	23.892 ± 7.131	22.189 ± 6.844	24.741 ± 7.144	14.040 ± 9.001
	13.694 ± 1.408	10.468 ± 0.663	16.832 ± 3.186	20.066 ± 1.192	23.807 ± 3.008	9.741 ± 1.698
InfoGAN	4.184 ± 0.693	7.700 ± 0.388	7.702 ± 2.137	8.213 ± 2.313	10.682 ± 1.933	5.653 ± 1.414
	4.707 ± 3.152	8.039 ± 3.353	8.623 ± 3.280	8.085 ± 3.983	12.106 ± 8.355	5.034 ± 3.126
	2.535 ± 0.277	5.789 ± 0.597	6.710 ± 2.561	6.351 ± 2.116	9.892 ± 3.821	2.207 ± 0.638
FisherGAN	1.823 ± 0.550	8.931 ± 0.997	5.493 ± 1.880	6.836 ± 1.186	10.790 ± 3.162	3.579 ± 2.350
	3.105 ± 1.345	11.792 ± 3.672	7.582 ± 2.777	8.348 ± 4.066	12.528 ± 3.745	3.895 ± 1.670
	2.768 ± 0.433	9.146 ± 0.449	7.666 ± 1.595	5.885 ± 0.885	10.648 ± 2.411	3.388 ± 0.929
ForwGAN	1.450 ± 0.426	8.835 ± 0.506	7.515 ± 2.029	5.828 ± 1.839	11.519 ± 3.067	3.431 ± 0.767
	2.634 ± 0.958	11.460 ± 3.812	9.553 ± 4.423	7.038 ± 2.696	11.161 ± 2.903	4.347 ± 1.576
	2.380 ± 0.413	9.478 ± 0.281	6.854 ± 2.344	4.799 ± 0.825	8.786 ± 1.829	3.220 ± 0.977
RevGAN	1.537 ± 0.456	8.915 ± 0.627	6.060 ± 1.804	4.752 ± 0.978	10.211 ± 3.843	2.889 ± 1.359
	2.358 ± 0.593	11.499 ± 3.694	9.208 ± 3.809	6.812 ± 2.425	10.693 ± 3.902	3.420 ± 1.207
	2.714 ± 0.336	9.645 ± 0.424	5.959 ± 1.767	5.355 ± 1.093	7.898 ± 1.725	3.665 ± 1.149
JSGAN	5.042 ± 0.709	10.094 ± 0.601	10.782 ± 2.684	9.379 ± 2.631	12.464 ± 1.795	4.090 ± 1.270
	5.276 ± 1.471	11.595 ± 2.963	12.447 ± 5.515	10.820 ± 2.957	14.465 ± 5.005	5.057 ± 2.490
	4.272 ± 0.908	10.478 ± 1.770	8.104 ± 1.919	9.201 ± 1.930	12.593 ± 2.252	3.932 ± 0.745
TVGAN	1.572 ± 0.425	9.026 ± 0.685	7.761 ± 1.566	6.326 ± 0.902	8.706 ± 1.955	2.868 ± 1.068
	3.024 ± 1.161	11.396 ± 3.091	8.336 ± 2.476	6.074 ± 2.376	10.191 ± 3.094	3.592 ± 1.347
	2.543 ± 0.421	9.273 ± 0.436	8.183 ± 2.106	6.407 ± 2.109	8.852 ± 2.168	3.529 ± 0.848
HellingerGAN	1.524 ± 0.341	8.882 ± 0.431	6.097 ± 1.572	5.846 ± 1.577	10.578 ± 2.485	3.140 ± 1.179
	2.694 ± 0.781	11.558 ± 3.325	8.278 ± 1.857	7.158 ± 2.250	10.792 ± 4.651	4.148 ± 1.731
	2.376 ± 0.346	9.982 ± 0.520	6.650 ± 1.888	6.172 ± 2.072	7.931 ± 1.591	2.925 ± 1.307
PearsonGAN	1.946 ± 0.439	9.315 ± 0.996	7.482 ± 3.103	5.619 ± 1.029	8.804 ± 1.810	2.961 ± 0.885
	3.243 ± 1.122	11.982 ± 4.928	7.740 ± 2.779	8.301 ± 3.403	13.876 ± 4.201	4.569 ± 1.747
	2.639 ± 0.246	9.390 ± 0.469	7.866 ± 1.670	7.304 ± 1.037	10.162 ± 2.995	3.465 ± 0.858

Table 5: Confidence intervals of the best average minimum Wasserstein distance across 20 trials for 1,000 samples (first row), 10,000 samples (second row), and 100,000 samples (third row) of the best hyperparameter settings for dimensionality $N = 128$.

D Diverse sets of hyperparameters can produce a best result

Model	Normal	Beta	Gumbel	Laplace	Exponential	Gamma
WGAN	[8, 3, 13]	[5, 3, 3]	[6, 6, 7]	[6, 5, 7]	[8, 11, 9]	[11, 8, 7]
	[14, 12, 10]	[5, 3, 4]	[5, 10, 5]	[14, 9, 9]	[10, 9, 10]	[11, 9, 7]
	[8, 11, 9]	[7, 6, 7]	[13, 11, 14]	[8, 10, 12]	[13, 13, 13]	[8, 10, 12]
WGANGP	[4, 8, 9]	[2, 3, 5]	[8, 7, 3]	[4, 5, 6]	[3, 6, 5]	[9, 4, 6]
	[5, 7, 7]	[3, 3, 3]	[6, 5, 2]	[4, 5, 4]	[8, 8, 10]	[11, 5, 6]
	[8, 6, 7]	[8, 8, 8]	[13, 13, 13]	[13, 12, 14]	[15, 12, 14]	[11, 10, 14]
NSGAN	[6, 5, 5]	[7, 4, 4]	[5, 7, 10]	[7, 4, 5]	[9, 4, 9]	[11, 8, 8]
	[9, 7, 8]	[9, 4, 4]	[5, 8, 7]	[8, 5, 6]	[9, 10, 12]	[11, 8, 12]
	[5, 6, 7]	[3, 5, 6]	[11, 10, 12]	[8, 11, 14]	[11, 9, 14]	[10, 9, 12]
MMGAN	[9, 8, 6]	[5, 4, 5]	[5, 4, 6]	[12, 11, 5]	[4, 5, 8]	[9, 7, 6]
	[8, 6, 4]	[3, 4, 5]	[6, 4, 5]	[7, 4, 2]	[5, 7, 10]	[5, 3, 7]
	[15, 8, 8]	[7, 3, 9]	[14, 6, 13]	[14, 9, 15]	[17, 11, 12]	[14, 15, 11]
RaGAN	[11, 8, 7]	[4, 6, 9]	[8, 10, 9]	[8, 9, 7]	[8, 5, 5]	[13, 12, 13]
	[11, 12, 10]	[6, 9, 10]	[9, 9, 13]	[13, 13, 11]	[8, 4, 4]	[12, 9, 11]
	[9, 5, 10]	[5, 8, 12]	[9, 14, 12]	[12, 10, 13]	[12, 12, 10]	[14, 12, 13]
LSGAN	[4, 9, 9]	[5, 5, 7]	[3, 5, 5]	[2, 6, 8]	[6, 5, 5]	[3, 7, 7]
	[5, 4, 3]	[5, 7, 6]	[3, 4, 5]	[2, 7, 4]	[8, 7, 7]	[4, 6, 8]
	[4, 6, 7]	[4, 8, 8]	[8, 10, 12]	[6, 11, 11]	[9, 12, 11]	[9, 7, 12]
DRAGAN	[12, 11, 7]	[9, 4, 5]	[9, 9, 6]	[9, 11, 9]	[9, 8, 4]	[14, 13, 7]
	[8, 6, 9]	[6, 4, 9]	[5, 4, 6]	[11, 6, 8]	[7, 4, 11]	[10, 7, 3]
	[13, 13, 5]	[6, 12, 5]	[10, 10, 10]	[12, 12, 9]	[12, 9, 9]	[11, 13, 7]
BEGAN	[11, 4, 5]	[8, 5, 5]	[7, 4, 3]	[6, 3, 3]	[6, 3, 6]	[4, 5, 3]
	[7, 3, 5]	[6, 6, 4]	[5, 5, 3]	[6, 4, 2]	[4, 3, 5]	[4, 2, 4]
	[7, 9, 4]	[12, 8, 9]	[17, 15, 4]	[13, 9, 3]	[15, 13, 15]	[9, 9, 3]
InfoGAN	[10, 10, 5]	[3, 4, 5]	[9, 5, 9]	[8, 10, 7]	[8, 4, 2]	[9, 4, 7]
	[7, 5, 3]	[4, 4, 5]	[8, 3, 8]	[5, 6, 5]	[9, 3, 4]	[12, 4, 9]
	[7, 7, 9]	[7, 10, 3]	[12, 11, 11]	[13, 13, 10]	[13, 14, 12]	[12, 4, 12]
FisherGAN	[6, 7, 5]	[9, 8, 7]	[6, 8, 9]	[8, 9, 7]	[10, 9, 13]	[10, 11, 9]
	[9, 12, 7]	[9, 10, 9]	[5, 8, 6]	[9, 9, 7]	[10, 7, 8]	[10, 10, 12]
	[7, 6, 10]	[4, 6, 6]	[12, 10, 13]	[10, 11, 10]	[11, 13, 15]	[11, 10, 9]
ForwGAN	[5, 6, 6]	[7, 9, 11]	[6, 6, 6]	[7, 6, 7]	[4, 10, 12]	[8, 7, 10]
	[8, 8, 8]	[8, 6, 7]	[3, 6, 4]	[9, 7, 7]	[11, 12, 9]	[11, 8, 10]
	[5, 6, 6]	[6, 8, 6]	[12, 14, 13]	[10, 11, 8]	[11, 11, 13]	[9, 11, 10]
RevGAN	[5, 8, 9]	[5, 10, 7]	[7, 7, 8]	[5, 5, 5]	[6, 11, 9]	[9, 8, 8]
	[8, 9, 9]	[9, 5, 7]	[4, 5, 3]	[7, 8, 7]	[11, 8, 10]	[9, 13, 12]
	[7, 7, 8]	[7, 8, 6]	[11, 11, 13]	[10, 11, 10]	[12, 10, 14]	[11, 12, 9]
JSGAN	[9, 11, 11]	[6, 6, 7]	[10, 11, 11]	[9, 7, 7]	[7, 5, 7]	[11, 6, 12]
	[6, 10, 8]	[5, 5, 5]	[9, 9, 9]	[9, 9, 8]	[6, 4, 10]	[12, 10, 13]
	[13, 12, 10]	[4, 9, 11]	[12, 10, 12]	[12, 14, 16]	[12, 11, 11]	[12, 10, 13]
TVGAN	[7, 6, 7]	[7, 7, 9]	[7, 8, 9]	[9, 8, 8]	[10, 11, 11]	[10, 11, 9]
	[9, 11, 12]	[10, 10, 9]	[9, 6, 5]	[9, 6, 9]	[7, 9, 10]	[11, 12, 11]
	[7, 7, 7]	[5, 5, 7]	[11, 10, 15]	[9, 9, 9]	[12, 12, 10]	[11, 8, 11]
HellingerGAN	[7, 5, 5]	[7, 8, 8]	[6, 6, 4]	[4, 4, 5]	[7, 8, 6]	[11, 9, 11]
	[9, 6, 9]	[6, 6, 4]	[5, 4, 3]	[7, 5, 6]	[9, 6, 6]	[11, 8, 8]
	[7, 6, 8]	[5, 5, 9]	[11, 10, 13]	[10, 11, 9]	[12, 14, 14]	[10, 11, 9]
PearsonGAN	[9, 9, 8]	[6, 8, 8]	[6, 7, 6]	[6, 8, 7]	[8, 6, 10]	[11, 9, 11]
	[13, 9, 10]	[8, 9, 8]	[8, 8, 10]	[11, 13, 11]	[10, 7, 10]	[14, 10, 11]
	[7, 6, 9]	[6, 11, 6]	[11, 9, 13]	[10, 11, 12]	[9, 12, 16]	[11, 11, 12]

Table 6: Number of unique hyperparameter settings that yielded a best result out of 20 trials for 1,000 samples (first entry of each row), 10,000 samples (second entry), and 100,000 samples (third entry) with respect to Kullback-Leibler divergence (first row), Jensen-Shannon divergence (second row), and Wasserstein distance (third row) for dimensionality $N = 128$. The maximum value for any given entry would be 20 provided that a diferent hyperparameter output the minimum performance for every trial conducted, and the minimum value would be 1 if only one hyperparameter setting always outperformed the rest. Dimensionalities $N \in \{16, 32, 64\}$ yielded similar results.

E Some GANs are more robust to hyperparameter changes than others

Model	Jensen-Shannon	Kullback-Leibler	Wasserstein Distance	Total (% of max)
WGAN	167	215	346	728 (1.40%)
WGANGP	115	146	492	753 (1.45%)
NSGAN	43	64	221	328 (0.63%)
MMGAN	34	127	416	577 (1.11%)
RaGAN	80	58	69	207 (0.40%)
LSGAN	73	120	386	579 (1.12%)
DRAGAN	126	138	380	644 (1.24%)
BEGAN	54	137	352	543 (1.05%)
InfoGAN	127	179	366	672 (1.30%)
FisherGAN	42	38	159	239 (0.46%)
ForwGAN	45	51	166	262 (0.51%)
RevGAN	43	44	137	224 (0.43%))
JSGAN	72	31	179	282 (0.54%)
TVGAN	38	47	126	211 (0.41%)
HellingerGAN	41	57	176	274 (0.53%)
PearsonGAN	32	42	165	239 (0.46%)

Table 7: The number of hyperparameter settings that yielded a minimum average performance that fell within the confidence interval of the best average performance across all 16 models, three evaluation metrics, six distributions, 15 hyperparameter settings, four dimensionalities, and three sample sizes. The last column indicates the sum of the values to its left followed by this number divided by 51,840 (the maximum possible number in this table) to its right, rounded up to the nearest hundredth.

F Model specifications

	$d = 16$	$d = 32$	$d = 64$	$d = 128$
$h = 32$	[1,393 - 1,821 - 1,888]	[2,433 - 2,861 - 3,456]	[4,513 - 4,941 - 6,592]	[8,673 - 9,101 - 12,864]
$h = 64$	[3,281 - 4,125 - 4,256]	[5,345 - 6,189 - 7,360]	[9,473 - 10,317 -13,568]	[17,729 - 18,573 - 25,984]
$h = 128$	[8,593 - 10,269 - 10,528]	[12,705 - 14,381 - 16,704]	[20,929 - 22,605 - 29,056]	[37,377 - 39,053 - 53,760]
$h = 256$	[25,361 - 28,701 - 29,216]	[33,569 - 36,909 - 41,536]	[49,985 - 53,325 - 66,176]	[82,817 - 86,157 - 115,456]
$h = 512$	[83,473 - 90,141 - 91,168]	[99,873 - 106,541 - 115,776]	[132,673 - 139,341 - 164,992]	[198,273 - 204,941 - 263,424]

Table 8: Number of model parameters for data dimensionality d and hidden dimension size h. The first entries are for all models except InfoGAN and BEGAN, the second entries are for InfoGAN, and the third are for BEGAN. All architectures consisted of four feedforward neural network layers in total: two for G and two for D. Since InfoGAN used latent variables as inputs to D and BEGAN's D was an autoencoder, they necessitated slightly more parameters. We did not observe these differences to give neither InfoGAN nor BEGAN any significant advantage over other models.