Effectively Unbiased FID and Inception Score and where to find them

Min Jin Chong and David Forsyth
University of Illinois at Urbana-Champaign
{mchong6, daf}@illinois.edu

Abstract

This paper shows that two commonly used evaluation metrics for generative models, the Fréchet Inception Distance (FID) and the Inception Score (IS), are biased – the expected value of the score computed for a finite sample set is not the true value of the score. Worse, the paper shows that the bias term depends on the particular model being evaluated, so model A may get a better score than model B simply because model A's bias term is smaller. This effect cannot be fixed by evaluating at a fixed number of samples. This means all comparisons using FID or IS as currently computed are unreliable.

We then show how to extrapolate the score to obtain an effectively bias-free estimate of scores computed with an infinite number of samples, which we term $\overline{FID}_\infty$ and $\overline{IS}_\infty$. In turn, this effectively bias-free estimate requires good estimates of scores with a finite number of samples. We show that using Quasi-Monte Carlo integration notably improves estimates of FID and IS for finite sample sets. Our extrapolated scores are simple, drop-in replacements for the finite sample scores. Additionally, we show that using low discrepancy sequence in GAN training offers small improvements in the resulting generator. The code for calculating $\overline{FID}_\infty$ and $\overline{IS}_\infty$ is at https://github.com/mchong6/FID_IS_infinity.

1. Introduction

Deep Generative Models have been used successfully to generate hyperrealistic images [18, 19, 7], map images between domains in an unsupervised fashion [42, 23], and generate images from text [40, 39]. Despite their widespread adoption, a simple, consistent, and human-like evaluation of generative models remain elusive, with multiple ad-hoc heuristics having their own faults. The Fréchet Inception Distance (FID) [15] was shown to have a high bias [4]; Inception Score (IS) [33] does not account for intra-class diversity and has further been shown to be suboptimal [3]; HYPE [41] requires human evaluation which makes large scale evaluation difficult; Kernel Inception Distance (KID) [4] has not been widely adopted, likely because of its relatively high variance [32]. Write FID_N and IS_N for FID and IS computed with N generated samples. Evaluation procedures with FID vary: some authors use FID_{50k} [7, 19, 18] while others use FID_{10k} [37, 25].

In this paper, we show that both FID and Inception Scores are biased differently depending on the generator. Biased estimators are often preferred over unbiased estimators for the reason of efficiency (a strong application example is the photon map in rendering [16]). In this case however, the bias is intolerable because for both FID and IS, the bias is a function both of N and the generator being tested. This means that we cannot compare generators because each has a different bias term (it is not sufficient to fix N, a procedure described in [4]). To fix this, we propose an extrapolation procedure on FID_N and IS_N to obtain an effectively unbiased estimate $\overline{\text{FID}}_\infty$ and $\overline{\text{IS}}_\infty$ (the estimate when evaluated with an unlimited number of generated images). In addition, both FID_∞ and IS_∞ are best estimated with low variance estimates of FID_N and IS_N. We show that Quasi-Monte Carlo Integration offers useful variance reduction in these estimates. The result is a simple method for unbiased comparisons between models. Conveniently, $\overline{\text{FID}}_\infty$ is a drop-in replacement for FID_N; $\overline{\text{IS}}_\infty$ for Inception Score. Our main contributions are as follows:

1. We show that FID_N and IS_N are biased, and cannot be used to compare generators.
2. We show that extrapolation of FID_N is reliable, and show how to obtain $\overline{\text{FID}}_\infty$ which is an effectively unbiased estimate of FID. Using Quasi-Monte Carlo integration methods yields better estimates of $\overline{\text{FID}}_\infty$.
3. We show the same for Inception Score and obtain $\overline{\text{IS}}_\infty$, an effectively unbiased estimate of Inception Score.
4. We show that Quasi-Monte Carlo integration methods offer small improvements in GAN training.

All figures are best viewed in color and high resolution.

2. Background

2.1. Fréchet Inception Distance

To compute Fréchet Inception Distance, we pass generated and true data through an ImageNet [9] pretrained Inception V3 [36] model to obtain visually relevant features. Let (M_t, C_t) and (M_g, C_g) represent the mean and covariance of the true and generated features respectively, then compute

$$\text{FID} = ||M_t - M_g||_2^2 + Tr(C_t + C_g - 2(C_t C_g)^{\frac{1}{2}}) \quad (1)$$

FID seems to correspond well with human judgement of image quality and diversity [38].

2.2. Inception Score

Write $g(z)$ for an image generator to be evaluated, y for a label, $p(y|x)$ for the posterior probability of a label computed using Inception V3 model on image x, $p(y) = \int p(y|g(z))dz$ for the marginal class distribution, and $\mathbb{D}(p \parallel q)$ for the KL-divergence between two probability distributions p and q. The Inception Score for a generator is

$$\exp\left[\mathbb{E}_{z\sim p(z)}[\mathbb{D}(p(y|g(z)) \parallel p(y))]\right] \quad (2)$$

Notice that the marginal class distribution is estimated using the same samples. This is important in our proof that IS is biased. The Inception Score takes into account two properties. 1) Images of meaningful objects should have a conditional label distribution of low entropy. 2) The marginals $p(y)$ should have high entropy if a model is able to generate varied images. A model that satisfy both properties will have a high IS.

2.3. Monte Carlo and Quasi-Monte Carlo Methods

The mean and covariance used in estimating FID_N are Monte Carlo estimates of integrals (the relevant expectations). The terms M_t and C_t computed on true images are not random, as proper comparisons fix the set of true images used. However, the terms M_g and C_g are random – if one uses different samples to evaluate these terms, one gets different values. A Monte Carlo (MC) estimate of an integral $\int h(x)p(x)dx$ whose true value is $I(h)$, made using N IID samples, yields $\hat{I} = I + \xi$, where

$$\mathbb{E}[\xi] = 0 \text{ and } \text{var}(\xi) = \frac{C(h)}{N} \quad (3)$$

where $C(h) \geq 0$ is $\int (h(x) - I(h))^2 p(x)dx$ [6]. Note the value of C is usually very hard to estimate directly, but C is non-negative and depends strongly on the function being integrated. A key algorithmic question is to identify procedures that result in lower variance estimates of the integral. Paskov [30] showed that Quasi-Monte Carlo Method

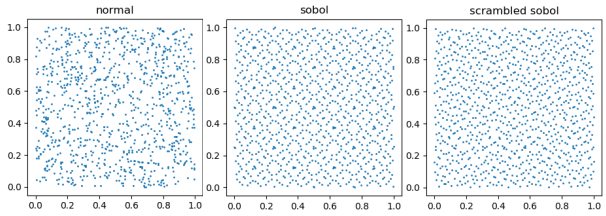


Figure 1: 2d scatter plots of 1000 random points vs Sobol and Scrambled Sobol points. Sobol sequences give us evenly spaced samples while random sampling results in clusters and empty spaces due to their IID property.

(QMC) with low-discrepancy sequences such as Sobol [35] and Halton [14] sequences gave a convergence of up to 5 times faster than MC with lower error rates. Both MC and QMC approximate

$$\int_{[0,1]^d} f(u)du \approx \frac{1}{N}\sum_{i=1}^{N} f(x_i) = \hat{I} \quad (4)$$

For MC estimates, x_i are IID samples from uniform distribution on the unit box; for QMC, x_i come from a deterministic quasi-random sequence. QMC can give faster convergence (close to $O(N^{-1})$, compared to MC's $O(N^{-0.5})$ [1]) and lower variance. This is because IID samples tend to be unevenly spaced (see "gaps" and "clusters" in Figure 1). QMC points are not independent, and so can be evenly spaced. A standard QMC construction is the Sobol sequence (review in Dick *et al.* [11]), which forms successively finer uniform partitions of each dimension and then reorders the coordinates to ensure a good distribution.

3. Evaluating generative models with FID

3.1. FID_N is Biased

Now consider some function G of a Monte Carlo integral $I(h)$, where G is sufficiently smooth. We have

$$G(\hat{I}) = G(I + \xi) \approx G(I) + \xi G^{'}(I) + \xi^2 \frac{G^{''}(I)}{2} + O(\xi^3) \quad (5)$$

so that

$$\mathbb{E}\left[G(\hat{I})\right] = G(I) + \frac{K}{N} + O(1/N^2) \quad (6)$$

where $K = C(h)\frac{G^{''}(I)}{2}$ and $\frac{K}{N}$ is bias.

Consider an estimate of FID, estimated with N samples. The terms M_g and C_g are estimated with an MC integrator, so the estimate must have a bias of $\frac{C_F}{N} + O(1/N^2)$. Note that C_F must *depend on the generator* g (section 2.3). Binkowski *et al.* [4] note that comparing two generators with different N is unreliable due to bias and that there may

be an effect that depends on the generator (but show no evidence). Experiment confirms that (a) FID_N is biased and (b) the bias depends on the generator (Figure 2).

3.2. $\overline{\text{FID}}_\infty$ as an Effectively Unbiased Estimate

The bias in FID_N vanishes for $N \to \infty$. Figure 2 suggests that the $O(1/N^2)$ terms are small for practical N, so we can extrapolate in $1/N$ to obtain $\overline{\text{FID}}_\infty$ (an estimate of the value of FID_∞). While $\overline{\text{FID}}_\infty$ could be still be biased by the higher order terms in the FID_N bias, our experiments suggest these are very small (line fits are good, see Figure 4). Thus, the bias and its dependence on the generator are small and $\overline{\text{FID}}_\infty$ is effectively unbiased. While Appendix D.3 of [4] implies there is no estimator of the FID that is unbiased for all distributions for a sample size N, our construction removes the very substantial dependence of the bias term on the generator, and so enables comparisons.

However, our extrapolation accuracy depends on the variance of our FID estimates. The estimates are a smooth function G of a Monte Carlo integral I. From section 3.1,

$$G(\hat{I}) = G(I) + \frac{K}{N} + O(1/N^2) \tag{7}$$

where K depends on C and the first derivative of G, and so

$$\text{var}(G(\hat{I})) = \frac{K_1}{N} + O(1/N^2) \tag{8}$$

where K_1 depends on C and first and second derivatives of G. Notice that this means that an integrator that yields a lower bias estimate of $G(\hat{I})$ will yield a lower variance estimate too for our case (where G is monotonic in I). This allows us to identify the integrator to use — we can find an integrator that yields low variance estimates of $G(I)$ by looking for one that yields the lowest mean value of $G(\hat{I})$. For FID, the best integrator to use is the one that yields the *lowest value* of estimated FID, and for IS, the one that yields the *highest value* of estimated IS for a given generator.

Quasi-Monte Carlo methods use low discrepancy sequences to estimate an integral. The Koksma-Hlawka inequality [28] gives that

$$\mid I_g - \hat{I} \mid \leq V[f \circ g] D_N^* \tag{9}$$

where $V[f \circ g]$ depends on the function to be integrated and is hard to determine, and D_N^* is the discrepancy of the sequence. It is usually difficult to estimate discrepancy, but for Sobol sequences it is $O((logN)^d N^{-1})$ where d is the number of dimensions; for random sequence is $O((\log\log N/N)^{0.5})$ [28]. As a result, integral estimates with low discrepancy sequences tend to have lower error for the same number of points, though dimension effects can significantly mitigate this improvement. Note that the reduced variance of Sobol sequence estimates manifests as reduced bias (smaller FID_N and larger IS_N) and variance in Table 1. In consequence, Sobol sequence results in better integrators.

Randomized Sobol sequences: it is useful to get multiple estimates of the integrals for an FID evaluation because this allows us to estimate the variance of the QMC which helps us construct approximate confidence intervals for the integral. However, low-discrepancy sequences such as the Sobol sequence are deterministic. One way to reintroduce randomness into QMC is to scramble the base digits of the sequence [29]. The resulting sequence will still have a QMC structure and the expectation of the integral remains the same.

3.3. IS_N is Biased

We show the log Inception Score is negatively biased, with a bias term that depends on the generator. Because the exponent is monotonic and analytic, this means the Inception Score is also biased negative with bias depends on the generator. Assuming we have N samples, $X_N = \{x_1, \ldots, x_N\}$ with two classes, let $\hat{P}_{1N} = \frac{1}{N}\sum_i p(1|x_i)$ and $p_{1i} = p(1|x_i)$. The log Inception Score over the samples is

$$\begin{aligned}
&\frac{1}{N}\Big[\sum_i p_{1i}\Big[\log p_{1i} - \log \hat{P}_{1N}\Big] \\
&+\sum_i (1-p_{1i})\Big[\log(1-p_{1i}) - \log(1-\hat{P}_{1N})\Big]\Big] \\
=&\frac{1}{N}\Big[\sum_i p_{1i}\log p_{1i} + (1-p_{1i})\log(1-p_{1i})\Big] \\
+&\frac{1}{N}\Big[-\log\hat{P}_{1N}\sum_i p_{1i} - \log(1-\hat{P}_{1N})\sum_i (1-p_{1i})\Big]
\end{aligned} \tag{10}$$

The first term is a Monte Carlo integral, and so is unbiased. The second term simplifies to the entropy of sample labels.

$$\frac{1}{N}\Big[-\hat{P}_{1N}\log\hat{P}_{1N} - (1-\hat{P}_{1N})\log(1-\hat{P}_{1N})\Big] \tag{11}$$

Let $G(u) = -u\log u - (1-u)\log(1-u)$. By Taylor Series,

$$\begin{aligned}
G(\hat{P}_{1N}) &= G(P_{1N} + \eta) \\
&\approx G(P_{1N}) + \eta\big(\log(1-P_{1N}) - \log P_{1N}\big) \\
&- \frac{\eta^2}{2}\left(\frac{1}{P_{1N}(1-P_{1N})}\right) + O(\eta^3)
\end{aligned} \tag{12}$$

where P_{1N} is the true integral. When we take expectation over samples, we have

$$G(\hat{P}_{1N}) = G(P_{1N}) - \frac{C}{2N}\left(\frac{1}{P_{1N}(1-P_{1N})}\right) + O\left(\frac{1}{N^2}\right) \tag{13}$$

because $\mathbb{E}[\eta] = 0$, $\mathbb{E}[\eta^2]$ as in section 3.1. Note that C must depend on the generator because $p(1|x)$ is shorthand for $p(1|g(z))$. The fact that entropy is convex yields a guaranteed *negative* bias in IS as the second derivative of a concave function is non-positive. The multiclass case follows.

All qualitative features of the analysis for FID are preserved. Particularly, bias depends on the generator (so no comparison with IS_N is meaningful); bias can be corrected by extrapolation (since IS_N is linear w.r.t $\frac{1}{N}$, see Figure 8); and improvements in integrator variance reduce IS_N bias and variance, see Table 1.

3.4. Uniform to Standard Normal Distribution

Low-discrepancy sequences are commonly designed to produce points in the unit hypercube. To make our work a direct drop-in replacement for current generators that use $\mathcal{N}(0, 1)$ as the prior for z, we explore two ways to transform a uniform distribution to a standard normal distribution. The main property we want to preserve after the transformation is the low-discrepancy of the generated points.

The Inverse Cumulative Distribution Function (ICDF), gives the value of the random variable such that the probability of it being less than or equal to that value is equal to the given probability. Specifically,

$$Q(p) = \sqrt{2}\,\text{erf}^{-1}(2p - 1), \quad p \in (0, 1) \tag{14}$$

where $Q(p)$ is the ICDF and erf is the error function. In our case, since our low-discrepancy sequence generates $\mathcal{U}[0, 1]$, we can treat them as probabilities and use $Q(p)$ to transform them into $\mathcal{N}(0, 1)$.

The Box-Muller transform (BM) [5]: Given $u \in (0, 1)^d$ where d is an even number, let u^{even} be the even-numbered components of u and u^{odd} be the odd-numbered components.

$$\begin{aligned} z_0 &= \sqrt{-2ln(u^{even})}\cos(2\pi u^{odd}) \\ z_1 &= \sqrt{-2ln(u^{even})}\sin(2\pi u^{odd}) \\ z &= (z_0, z_1) \end{aligned} \tag{15}$$

The computations for both methods are negligible, making them efficient for our use case. Okten [27] provided theoretical and empirical evidence that BM has comparable or lower QMC errors compared to ICDF. Our experiments include using both methods, which we dub Sobol_{BM} and Sobol_{Inv}. We show that both perform better than random sampling but generally, Sobol_{Inv} gives better estimates for FID_∞ and IS_∞ than Sobol_{BM}.

3.5. Training with Sobol Sequence

We explore training Generative Adversarial Networks (GANs) [12] with Sobol sequence. GANs are notorious for generating bad images at the tails of the normal distribution where the densities are poorly represented during training. There are several methods such as the truncation trick [7, 19] which avoids these tail regions to improve image quality at the cost of image diversity. We hypothesize that by using Sobol sequence during GAN training instead of normal sampling, the densities of the distribution will be better represented, leading to overall better generation quality. Furthermore, we can view training GANs as estimating an integral as it involves sampling a small batch of z and computing the unbiased loss estimate over it. Though we are choosing a small N (batch size in our case), using Quasi-Monte Carlo integration might still lead to a reduction in the variance of the loss estimate.

We note that training a GAN with Sobol sequence has been done before[1]. This effort failed because the high-dimensional Sobol points were not correctly generated and were not shuffled. We will describe a successful attempt to train a GAN with Sobol sequence in section 4.7.

4. Experiments

In our experiments, we find that

1. FID is linear with respect to $\frac{1}{N}$ and different generators have very different K, so that generators cannot be compared with FID_N for any finite N (section 4.1).
2. Using Sobol sequence integrators reliably results in lower bias (and so lower variance) in the estimated FID (section 4.2).
3. Extrapolating the value of FID_{100k} from smaller N compares very well with true estimates. Thus FID_∞ can be estimated effectively with low variance using Sobol points (section 4.3 and section 4.4).
4. FID_∞ can be estimated effectively for other models such as VAEs [21] too (section 4.5).
5. Inception Score behaves like FID but with negative bias. We can estimate $\overline{\text{IS}}_N$ accurately (section 4.6).
6. Training GANs with Sobol sequence yields better $\overline{\text{FID}}_\infty$ scores with lower variance across models (section 4.7).

Our experiments focus mainly on GANs as they are one of the most popular deep generative models today. We ran our evaluations on DCGAN [31], ProGAN [18], StyleGAN [19], and BigGAN [7]. For the implementation of Sobol sequence, we use QMC sampler from BoTorch [2].

We trained a DCGAN on 64×64 resolution CelebA [24] for 50 epochs using TTUR [15] with Adam Optimizer [20] and Spectral Normalization [26]. For ProGAN, we use a pretrained CelebA model for generating 1024×1024 resolution images. For StyleGAN, we use a pretrained Flickr-Faces-HQ model for generating 1024×1024 resolution im-

[1] https://github.com/deeptechlabs/sobol_noise_gan

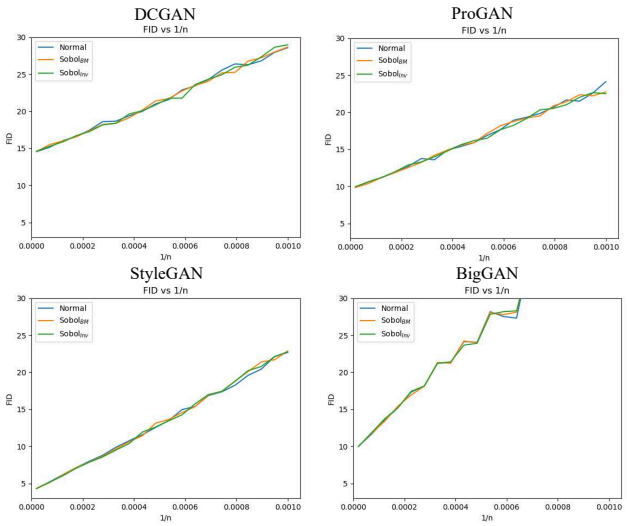


Figure 2: Plots of FID vs $\frac{1}{N}$ for various models with various sampling methods at the same scale. Each FID point corresponds to a single FID estimate. FIDs are linear with respect to $\frac{1}{N}$ across all experiments, with higher variance (more spikes) when N is small. Most importantly, the slopes, which corresponds to the K term of eq 7, are very different across models. Even though the models are not directly comparable since they generate different datasets, this shows different models have very different K terms. Comparisons of different models with FID_N at fixed N are unreliable because they are dominated by bias.

ages. We also evaluated on BigGan which is a conditional GAN. We use a pretrained ImageNet BigGAN model which generates 128×128 resolution images. We also train BigGAN on CIFAR10 [22] and use that for our evaluations.

4.1. FID_N Bias

Across different models, we compare FID at different values of $\frac{1}{N}$ and show that they have a linear relationship as seen in Figure 2. As expected from equation 8, when N is small, the variance of the FID is higher. Importantly, we observe that across different models, the slope varies significantly. The slope corresponds to the K term in equation 7 which contributes to the FID bias. In effect, the rankings between GANs are severely dependent on N as different GANs will have different biases that changes with N. This can even be seen in two models with the exact same architecture, Figure 3. Appendix D.2 of [4] also gives an empirical example of FID_N reliably giving the wrong ranking of models under somewhat realistic setting. There is no one N that works for every comparison as it depends on the K term of each model. *No comparison that uses FID_N is reliable*.

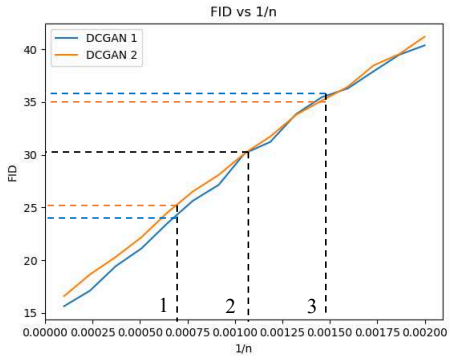


Figure 3: The choice of N used affects comparison severely. The graph compares the FID_N vs $\frac{1}{N}$ between two independently trained identical architecture DCGANs. At marker 1, DCGAN 1 is better than DCGAN 2; at marker 2 they are approximately the same; at marker 3 DCGAN 2 is better than DCGAN 1. This shows that comparisons between models of a fixed N are unreliable.

4.2. Evaluating with different sampling schemes

FID is commonly computed with 50k samples and IS with 5k samples. Recall section 3.2 establishes that an integrator that produces a lower bias estimate (which can't be observed) will also produce a lower variance estimate (which can be observed). Table 1 compares 50 runs each of FID_{50k} and IS_{5k} for a variety of models, estimated using IID normal samples or a Sobol sequence with either Box-Muller transform or ICDF. It is clear from the table that GAN evaluation should always use a low-discrepancy sequence, because evaluating with Sobol_{BM} and Sobol_{Inv} gives a better FID and Inception Score with lower standard deviations.

4.3. FID can be extrapolated

Using the property that FID is linear with respect to $\frac{1}{N}$, we test the accuracy of estimating FID_{100k} given only 50k images. We do that by first generating a pool of 50k images from the generator and randomly sampling them with replacement to compute 15 FIDs. We then fit a linear regression model over the points, which we can then use for extrapolating $\overline{\text{FID}}_{100k}$. Pseudo-code can be found in our Appendix.

We tried two ways of choosing the number of images to evaluate the FID on.

1. choosing over regular intervals of N
2. choosing over regular intervals of $\frac{1}{N}$

In total, for each of our test models, we run 6 different experiments, each for 50 runs. Three ways of sampling z (Normal sampling, Sobol_{BM}, Sobol_{Inv}) and two ways of choosing N for evaluation.

Computing FID values for N that are evenly spaced in

	Normal		Sobol$_{BM}$			Sobol$_{Inv}$		
Models	FID$_{50k}$	IS$_{5k}$	FID$_{50k}$	IS$_{5k}$	F Value	FID$_{50k}$	IS$_{5k}$	F Value
DCGAN	14.61 ± 0.0579	-	14.59 ± 0.0471	-	1.51	**14.58 ± 0.0439**	-	1.74
ProGAN	9.94 ± 0.0411	-	**9.94 ± 0.0384**	-	1.14	9.94 ± 0.0404	-	1.03
StyleGAN	4.33 ± 0.0413	-	4.33 ± 0.0406	-	1.03	**4.33 ± 0.0354**	-	1.36
BigGAN	9.94 ± 0.0564	92.96 ± 2.135	**9.92** ± 0.0576	92.89 ± 1.961	1.19	$9.93 \pm$ **0.0419**	**93.21 ± 1.640**	1.69
BigGAN (CIFAR10)	8.26 ± 0.0467	8.44 ± 0.1223	**8.25** ± 0.0455	**8.48** ± 0.1172	1.09	$8.26 \pm$ **0.0446**	$8.45 \pm$ **0.1018**	1.44

Table 1: Using Sobol sequences always give better FID and Inception Score (IS) along with lower standard deviations. The table shows FID$_{50k}$ (lower better) and IS$_{5k}$ (higher better) values of different models evaluated on Normal and Sobol sequences over 50 runs. The F value is the ratio between the variances of the Normal and Sobol Sequences (the higher it is, the more different their variances are). Bolded values indicate the best score or standard deviation. Better scores implies lower bias which implies lower integrator variance.

$\frac{1}{N}$ results in an even looking plot, but a weaker extrapolate, see Figure 4. This is because most estimates will be in the region with small N, which is noisier. This leads to a poor $\overline{\text{FID}}_{100k}$ estimate because the FIDs evaluated at those points have a high variance according to equation 8. Computing the score at regular intervals over N works better in practice. To ensure that the FIDs we calculate are reliable, we use at least 5k points.

From Figure 5, we can see that across all experiments, $\overline{\text{FID}}_{100k}$ is very accurate. Overall, normal random sampling gives a decent estimation but the variance of the estimate is higher compared to using Sobol sequence. Sobol$_{BM}$ has the lowest variance, however, its estimation is not as accurate. Sobol$_{Inv}$ overall gives the best result, giving us an accurate $\overline{\text{FID}}_{100k}$ estimate with low variance. This fits into our expectation as FIDs evaluated from Sobol sequence have lower variance, giving us a better line fit, resulting in a more accurate prediction.

More careful tuning of hyperparameters (total number of images and the number of FIDs to fit a line) could yield better $\overline{\text{FID}}_{100k}$ estimates.

4.4. FID$_\infty$

Since we showed that simple linear regression gives us good prediction accuracy for FID$_{100k}$, we can then extend to estimating FID$_\infty$. Following previous setup, we obtain $\overline{\text{FID}}_\infty$ estimate using 50k samples. Though we do not have the groundtruth FID$_\infty$, our $\overline{\text{FID}}_\infty$ estimates (Figure 6) have similar means across different sampling methods and have small variances. This together with our experiments in section 4.3 suggests our $\overline{\text{FID}}_\infty$ estimates are accurate.

4.5. FID$_\infty$ for VAE

Our FID results apply to any generative model. Our experiments on VAE shows the same linear property of FID, improved bias and variance with QMC, and successful extrapolations. For brevity, we show only the $\overline{\text{FID}}_\infty$ plots in Figure 7 for a vanilla VAE trained on 64×64 CelebA.

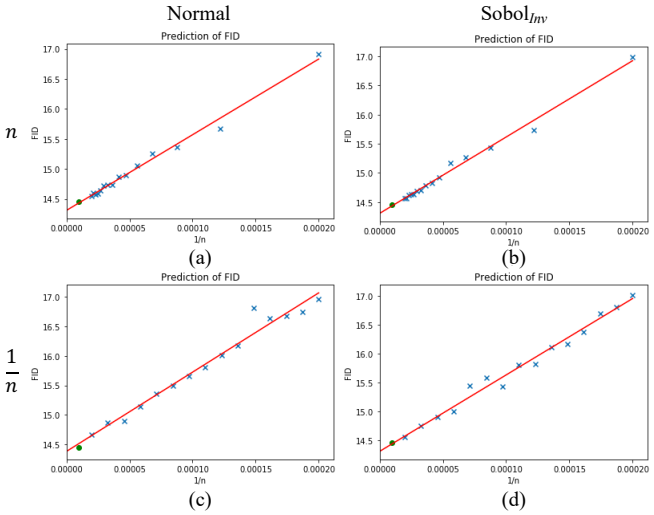


Figure 4: FID$_N$ estimates give very good line fits, especially with Sobol$_{Inv}$ (right), suggesting extrapolation will succeed (it does, see Fig 5). The figure shows the line fits for predicting FID$_{100k}$ for a random DCGAN. Green point is the target FID$_{100k}$ while the blue crosses are the FIDs we compute to fit a linear regression. The columns represents the sampler we use to generate images while the rows represent how we choose N to compute FID. **Row 1:** choose at regular intervals over N; **Row 2:** choose at regular intervals over $\frac{1}{N}$. For the normal sampler, there are more outliers and the prediction is not as accurate. Using Sobol$_{Inv}$ and computing FIDs at regular intervals over N (Figure **(b)**) give us better line fit for predicting FID$_{100k}$.

4.6. Estimating IS$_\infty$

Inception Score follows the same trend as FID, namely that it is linear with respect to $\frac{1}{N}$ (see Figure 8) and thus can be extrapolated to obtain $\overline{\text{IS}}_\infty$ estimate. However, it seems that the variance of IS$_N$ estimates varies greatly for differently generators, see Table 1. This results in larger variance in our $\overline{\text{IS}}_\infty$ estimate which QMC can help reduce, see Fig-

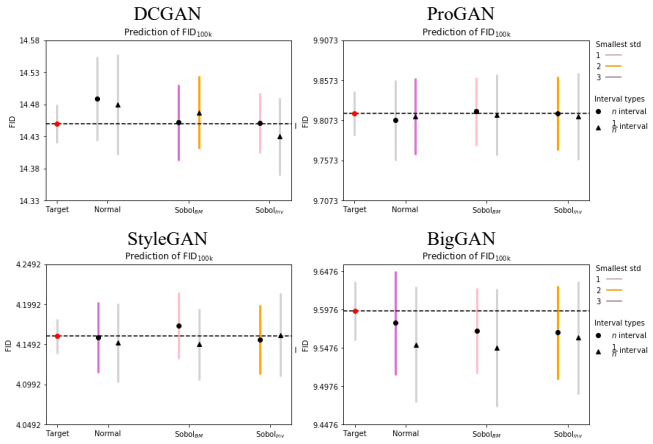


Figure 5: FID_{100k} predictions are highly accurate compared to groundtruth (horizontal line) with low variance using 50k images. This suggests predicting FID_∞ is sound. The figure shows error plots of $\overline{\text{FID}}_{100k}$ with y axis of the same scale. The point represent the mean, and the error bar the standard deviation over 50 runs. The far left point is the target FID_{100k} we are estimating. For each sampling method, we estimate FID_{100k} by fitting points over regular intervals over N (dots) or over $\frac{1}{N}$ intervals (triangles). We also color-tagged the lowest 3 standard deviations. Overall, Sobol_{Inv} with intervals over N perform the best, with good accuracy and low variance. Best viewed in color and high resolution.

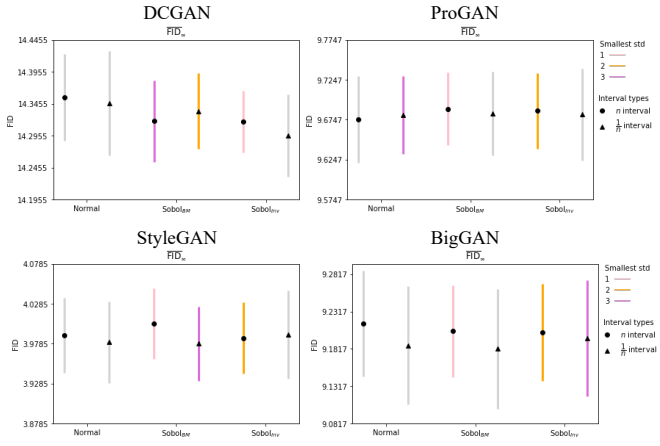


Figure 6: Error plots of predicting FID_∞ over 50 runs. $\overline{\text{FID}}_\infty$ have low variance and are consistent. This together with Figure 5 suggests that they are accurate. The plots follow the same markings as Figure 5.

ure 8. Figure 9 shows that the estimated $\overline{\text{IS}}_{100k}$ for BigGAN trained on ImageNet is very accurate with comparable variance to the actual IS_{100k} estimate. For CIFAR10 BigGAN,

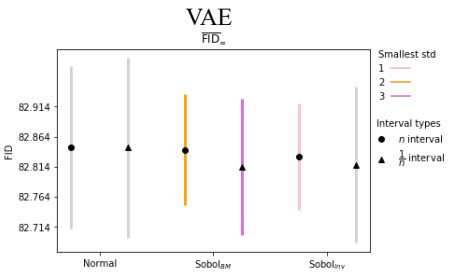
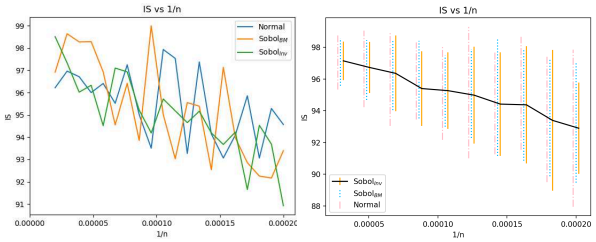


Figure 7: Error plots of predicting FID_∞ for a VAE over 50 runs. $\overline{\text{FID}}_\infty$ works regardless for model used, in this case for VAE. The plots follow the same markings as Figure 5.



(a) 1 split (b) 50 splits

Figure 8: IS_N is negatively biased. IS_N vs $\frac{1}{N}$ of BigGAN for 3 sampling methods where each point is an average over (a): 1 split, (b): 50 splits. For (b), error bars represent standard deviation and the line joins the IS_N estimates mean of Sobol_{Inv}. Unlike FID_N, IS_N increases with increasing N, suggesting negative bias. From (b) the variance of IS_N estimates from normal points are considerably higher than those of Sobol sequences, which is also evidenced by Table 1. Figure (b) is best viewed in color and high resolution.

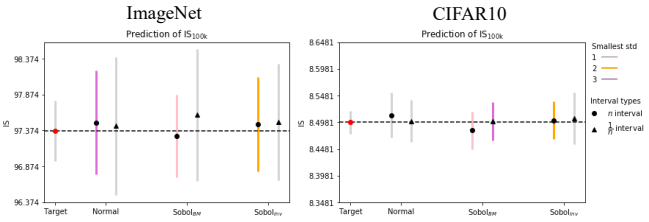
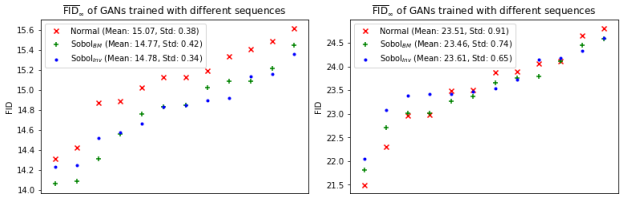


Figure 9: IS_{100k} predictions using 50k images are highly accurate with low variance. The figure shows error plots of $\overline{\text{IS}}_{100k}$ for BigGAN trained on ImageNet and CIFAR10 over 50 runs. Sobol_{Inv} gives the best accuracy with low standard deviations. The plots follow the same markings as Figure 5.

our $\overline{\text{IS}}_{100k}$ with Sobol_{Inv} is very accurate with low variance. In general, extrapolating with QMC works and we can get an effectively unbiased estimate $\overline{\text{IS}}_\infty$ with good accuracy and low variance.



64 x 64 128 x 128

Figure 10: Training GANs with Sobol sequence results in better or comparable $\overline{\text{FID}}_\infty$ and lower variance between training runs. The figures show sorted $\overline{\text{FID}}_\infty$ of 12 GANs trained with different sampling methods for two resolutions of CelebA. Each $\overline{\text{FID}}_\infty$ is an average of 50 calculations.

4.7. Training with Sobol sequence

We trained DCGAN on CelebA at both 64×64 and 128×128 resolution with the same setup as before. For GANs, we need two separate samplers for the generator and discriminator so that the "uniform" property of the sequence will not be split between them two. Since Sobol points are highly correlated with each other even with different scramblings, using two Sobol samplers will cause unstable GAN training. Instead, we cache 1×10^6 points for both samplers and shuffle them to break their correlation.

For each of the 3 sampling methods, we trained 12 models and we evaluate their $\overline{\text{FID}}_\infty$ score over 50 runs. For $\overline{\text{FID}}_\infty$, we use Sobol$_{Inv}$ with regular intervals over N. From Figure 10, the $\overline{\text{FID}}_\infty$ of GANs trained with Sobol sequences are generally lower at 64×64 and are comparable with normal sampling at 128×128. However, for both resolutions, GANs trained with Sobol$_{Inv}$ have significantly less $\overline{\text{FID}}_\infty$ variance between different runs compared to normal sampling. The improvements are consistent and essentially free as the computational overhead is negligible. We believe further experimentations with more models and datasets could yield interesting results.

5. Related works

Demystifying MMD GANs: Binkowski *et al.* [4] showed that there is no unbiased estimator for FID. However, the Stone–Weierstrass theorem allows arbitrarily good uniform approximation of functions on the unit interval by sufficiently high degree polynomials. Thus, while zero bias is unattainable, very small bias is not ruled out. Our $\overline{\text{FID}}_\infty$ and $\overline{\text{IS}}_\infty$ scores clearly display very small bias because (as the graph shows) the $\frac{1}{N}$ terms dominate higher order terms.

Debiasing via Importance Weighting: Grover *et al.* [13] reduce errors in MC estimates computed with augmented datasets by using a classifier to estimate importance weights. By doing so, they exhibit improved IS_N, FID_N, and KID_N scores. We believe the improvements are the result of increased effective sample size from the augmentation. However, in contrast to our work, they do not identify formal statistical bias in FID_N or IS_N, nor do they point out that the dependence of this bias on the generator makes comparisons at fixed N unreliable.

QMC Variational Inference: Buchholz *et al.* [8] propose using QMC to reduce the variance of the gradient estimators of Monte Carlo Variational Inference. In their Appendix D, they suggest using QMC for VAEs and GANs. However, they offer no explanations nor results from doing so.

HYPE: Note that [41] computes a correlation between HYPE and FID_N for generators trained on different ImageNet classes. Because the FID_N scores are biased, *with a bias that depends on the particular generator*, the correlations cannot be relied upon. It would be interesting to correlate FID_∞ with HYPE.

6. Future work

This paper serves as an introduction of using Quasi-Monte Carlo methods to estimate high dimensional integrals in the field of generative models for bias reduction. However, there has been a substantial amount of work in the area of estimating high dimensional integrals such as sparse grids [34], higher order scrambled digital nets [10], randomized lattice rules [17] which we have yet to touch upon. Furthermore, using a closed quasi-random sequence (where we know N beforehand) for evaluation could give us better error bounds on the integral [11]. We reserve these for our future work. Also, FID_∞ could well correlate with HYPE and we plan to investigate these correlations.

7. Best practices

7.1. For evaluating generators with FID or IS

Never compare generators with FID_N or IS_N; the comparisons are not reliable.

1. Use Sobol$_{Inv}$ to compute both FID_N and IS_N.
2. Using estimates obtained at regular intervals over N, extrapolate to get $\overline{\text{FID}}_\infty$ and $\overline{\text{IS}}_\infty$ estimates.
3. Repeat multiple times to get a variance estimate. This corresponds to how reliable the $\overline{\text{FID}}_\infty$ and $\overline{\text{IS}}_\infty$ estimates are.

7.2. For training GANs

We have moderate results to show that training with Sobol sequence results in better or comparable $\overline{\text{FID}}_\infty$ and lower variance across models. We believe that large-scale experiments should be done to validate the usefulness of training with Sobol sequence and that is left to future work.

References

[1] Søren Asmussen and Peter W Glynn. *Stochastic simulation: algorithms and analysis*, volume 57. Springer Science & Business Media, 2007. 2

[2] Maximilian Balandat, Brian Karrer, Daniel R. Jiang, Samuel Daulton, Benjamin Letham, Andrew Gordon Wilson, and Eytan Bakshy. BoTorch: Programmable Bayesian Optimization in PyTorch. *arxiv e-prints*, 2019. 4

[3] Shane Barratt and Rishi Sharma. A note on the inception score. *arXiv preprint arXiv:1801.01973*, 2018. 1

[4] Mikołaj Bińkowski, Dougal J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018. 1, 2, 3, 5, 8

[5] George EP Box. A note on the generation of random normal deviates. *Ann. Math. Stat.*, 29:610–611, 1958. 4

[6] Phelim P Boyle. Options: A monte carlo approach. *Journal of financial economics*, 4(3):323–338, 1977. 2

[7] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018. 1, 4

[8] Alexander Buchholz, Florian Wenzel, and Stephan Mandt. Quasi-monte carlo variational inference. *arXiv preprint arXiv:1807.01604*, 2018. 8

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2

[10] Josef Dick et al. Higher order scrambled digital nets achieve the optimal rate of the root mean square error for smooth integrands. *The Annals of Statistics*, 39(3):1372–1398, 2011. 8

[11] Josef Dick, Frances Y Kuo, and Ian H Sloan. High-dimensional integration: the quasi-monte carlo way. *Acta Numerica*, 22:133–288, 2013. 2, 8

[12] 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. 4

[13] Aditya Grover, Jiaming Song, Alekh Agarwal, Kenneth Tran, Ashish Kapoor, Eric Horvitz, and Stefano Ermon. Bias correction of learned generative models using likelihood-free importance weighting. *arXiv preprint arXiv:1906.09531*, 2019. 8

[14] John H Halton. Algorithm 247: Radical-inverse quasi-random point sequence. *Communications of the ACM*, 7(12):701–702, 1964. 2

[15] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, pages 6626–6637, 2017. 1, 4

[16] Henrik Wann Jensen. Global illumination using photon maps. In *Rendering Techniques' 96*, pages 21–30. Springer, 1996. 1

[17] Stephen Joe. Randomization of lattice rules for numerical multiple integration. *Journal of Computational and Applied Mathematics*, 31(2):299–304, 1990. 8

[18] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 1, 4

[19] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2019. 1, 4

[20] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[21] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 4

[22] Alex Krizhevsky et al. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. 5

[23] Ming-Yu Liu, Xun Huang, Arun Mallya, Tero Karras, Timo Aila, Jaakko Lehtinen, and Jan Kautz. Few-shot unsupervised image-to-image translation. *arXiv preprint arXiv:1905.01723*, 2019. 1

[24] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015. 4

[25] Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are gans created equal? a large-scale study. In *Advances in neural information processing systems*, pages 700–709, 2018. 1

[26] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018. 4

[27] Giray Ökten and Ahmet Göncü. Generating low-discrepancy sequences from the normal distribution: Box–muller or inverse transform? *Mathematical and Computer Modelling*, 53(5-6):1268–1281, 2011. 4

[28] Art B Owen. Quasi-monte carlo sampling. 3

[29] Art B Owen. Randomly permuted (t, m, s)-nets and (t, s)-sequences. In *Monte Carlo and quasi-Monte Carlo methods in scientific computing*, pages 299–317. Springer, 1995. 3

[30] Spassimir Paskov and Joseph F Traub. Faster valuation of financial derivatives. 1996. 2

[31] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 4

[32] Suman Ravuri and Oriol Vinyals. Seeing is not necessarily believing: Limitations of biggans for data augmentation. 2019. 1

[33] 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. 1

[34] Sergei Abramovich Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. In *Doklady Akademii Nauk*, volume 148, pages 1042–1045. Russian Academy of Sciences, 1963. 8

[35] Il'ya Meerovich Sobol'. On the distribution of points in a cube and the approximate evaluation of integrals. *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, 7(4):784–802, 1967. 2

[36] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 2

[37] Ilya Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schoelkopf. Wasserstein auto-encoders. *arXiv preprint arXiv:1711.01558*, 2017. 1

[38] Qiantong Xu, Gao Huang, Yang Yuan, Chuan Guo, Yu Sun, Felix Wu, and Kilian Weinberger. An empirical study on evaluation metrics of generative adversarial networks. *arXiv preprint arXiv:1806.07755*, 2018. 2

[39] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1316–1324, 2018. 1

[40] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stackgan++: Realistic image synthesis with stacked generative adversarial networks. *IEEE transactions on pattern analysis and machine intelligence*, 41(8):1947–1962, 2018. 1

[41] Sharon Zhou, Mitchell Gordon, Ranjay Krishna, Austin Narcomey, Durim Morina, and Michael S Bernstein. Hype: Human eye perceptual evaluation of generative models. *arXiv preprint arXiv:1904.01121*, 2019. 1, 8

[42] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 1