A Domain Agnostic Measure for Monitoring and Evaluating GANs

Paulina Grnarova[*]
ETH Zurich

Kfir Y. Levy
Technion-Israel Institute of Technology

Aurelien Lucchi
ETH Zurich

Nathanaël Perraudin
Swiss Data Science Center

Ian Goodfellow

Thomas Hofmann
ETH Zurich

Andreas Krause
ETH Zurich

Abstract

Generative Adversarial Networks (GANs) have shown remarkable results in modeling complex distributions, but their evaluation remains an unsettled issue. Evaluations are essential for: (i) relative assessment of different models and (ii) monitoring the progress of a single model throughout training. The latter cannot be determined by simply inspecting the generator and discriminator loss curves as they behave non-intuitively. We leverage the notion of duality gap from game theory to propose a measure that addresses both (i) and (ii) at a low computational cost. Extensive experiments show the effectiveness of this measure to rank different GAN models and capture the typical GAN failure scenarios, including mode collapse and non-convergent behaviours. This evaluation metric also provides meaningful monitoring on the progression of the loss during training. It highly correlates with FID on natural image datasets, and with domain specific scores for text, sound and cosmology data where FID is not directly suitable. In particular, our proposed metric requires no labels or a pretrained classifier, making it domain agnostic.

1 Introduction

In recent years, a large body of research has focused on practical and theoretical aspects of Generative adversarial networks (GANs) [9]. This has led to the development of several GAN variants [24, 2] as well as some evaluation metrics such as FID or the Inception score that are both data-dependent and dedicated to images. A *domain independent* quantitative metric is however still a key missing ingredient that hinders further developments.

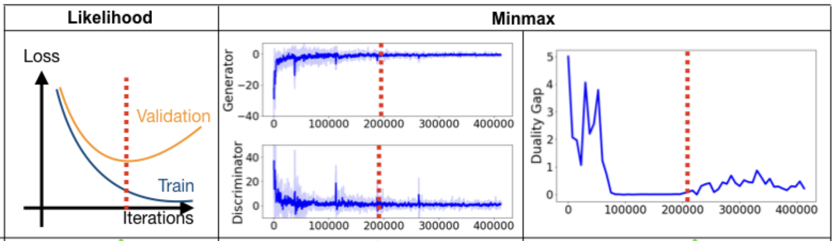
One of the main reasons behind the lack of such a metric originates from the nature of GANs that implement an adversarial game between two players, namely a generator and a discriminator. Let us denote the data distribution by $p_{\text{data}}(\mathbf{x})$, the model distribution by $p_{\mathbf{u}}(\mathbf{x})$ and the prior over latent variables by $p_{\mathbf{z}}$. A probabilistic discriminator is denoted by $D_{\mathbf{v}} : \mathbf{x} \mapsto [0; 1]$ and a generator by $G_{\mathbf{u}} : \mathbf{z} \mapsto \mathbf{x}$. The GAN objective is:

$$\min_{\mathbf{u}} \max_{\mathbf{v}} M(\mathbf{u}, \mathbf{v}) = \frac{1}{2}\mathbb{E}_{\mathbf{x}\sim p_{\text{data}}} \log D_{\mathbf{v}}(\mathbf{x}) + \frac{1}{2}\mathbb{E}_{\mathbf{z}\sim p_{\mathbf{z}}} \log(1 - D_{\mathbf{v}}(G_{\mathbf{u}}(\mathbf{z}))). \tag{1}$$

Each of the two players tries to optimize their own objective, which is exactly balanced by the loss of the other player, thus yielding a two-player zero-sum minimax game. The minimax nature of the objective and the use of neural networks as players make the process of learning a generative model challenging. We focus our attention on two of the central open issues behind these difficulties and how they translate to a need for an assessment metric.

[*] Correspondence to paulina.grnarova@inf.ethz.ch

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.



	Likelihood	Minmax	
(non-)convergence detection	✓	✗	✓
Stopping criteria	✓	✗	✓
Hyperparameter tuning	✓	✗	✓
Model ranking	✓	✗	✓
Meaningful curve	✓	✗	✓

Figure 1: Comparison of information obtained by different metrics for likelihood and minmax-based models. The red dashed line corresponds to the optimal point for stopping the training.

i) Convergence metric The need for an adequate convergence metric is especially relevant given the difficulty of training GANs: current approaches often fail to converge [31] or oscillate between different modes of the data distribution [21]. The ability of reliably detecting non-convergent behavior has been pointed out as an open problem in many previous works, e.g., by [20] as a stepping stone towards a deeper analysis as to which GAN variants converge. Such a metric is not only important for driving the research efforts forward, but from a practical perspective as well. Deciding when to stop training is difficult as the curves of the discriminator and generator losses oscillate (see Fig. 1) and are non-informative as to whether the model is improving or not [2]. This is especially troublesome when a GAN is trained on non-image data in which case one might *not* be able to use visual inspection or FID/Inception scores as a proxy.

ii) Evaluation metric Another key problem we address is the relative comparison of the learned generative models. While several evaluation metrics exist, there is no clear consensus regarding which metric is the most appropriate. Many metrics achieve reasonable discriminability (i.e., ability to distinguish generated samples from real ones), but also tend to have a high computational cost. Some popular metrics are also specific to image data. We refer the reader to [3] for an in-depth discussion of the merits and drawbacks of existing evaluation metrics.

In more traditional likelihood-based models, the train/test curves do address the problems raised in **i)** and **ii)**. For GANs, the generator/discriminator curves (see Fig. 1) are however largely uninformative due to the minimax nature of GANs where both players can undo each other's progress.

In this paper, we leverage ideas from game theory to propose a simple and computationally efficient metric for GANs. Our approach is to view GANs as a zero-sum game between a generator G and discriminator D. From this perspective, "solving" the game is equivalent to finding an *equilibrium*, i.e., a pair (G^*, D^*) such that no side may increase its utility by unilateral deviation. A natural metric for measuring the sub-optimality (w.r.t. an equilibrium) of a given solution (G, D) is the *duality gap* [33, 22]. We therefore suggest to use it as a metric for GANs akin to a test loss in the likelihood case (See Fig. 1 - duality gap[2]).

There are several important issues that we address in order to make the *duality gap* an appropriate and practical metric for GANs. Our contributions include the following:

- We show that the *duality gap* allows to assess the similarity between the generated data and true data distribution (see Theorem 1).
- We show how to appropriately estimate the *duality gap* in the typical machine learning scenario where our access to the GAN learning objective is only through samples.
- We provide a computationally efficient way to estimate the *duality gap* during training.
- In scenarios where one is interested in assessing the quality of the learned generator, we show how to use a related metric – the minimax loss – that takes only the generator into consideration in order to detect mode collapse and measure sample quality.

[2] The curves are obtained for a progressive GAN trained on CelebA

- We extensively demonstrate the effectiveness of these metrics on a range of datasets, GAN variants and failure modes. Unlike the FID or Inception score that require labelled data or a domain dependent classifier, our metrics are *domain independent* and do not require labels.

Related work. While several evaluation metrics have been proposed [31, 30, 12, 18], previous research has pointed out various limitations of these metrics, thus leaving the evaluation of GANs as an unsettled issue [20]. Since the data log-likelihood is commonly used to train generative models, it may appear to be a sensible metric for GANs. However, its computation is often intractable and [32] also demonstrate that it has severe limitations as it might yield low visual quality samples despite of a high likelihood. Perhaps the most popular evaluation metric for GANs is the inception score [31] that measures both diversity of the generated samples and discriminability. While diversity is measured as the entropy of the output distribution, the discriminability aspect requires a pretrained neural network to assign high scores to images close to training images. Various modifications of the inception score have been suggested. The Frechet Inception Distance (FID) [12] models features from a hidden layer as two multivariate Gaussians for the generated and true data. However, the Gaussian assumption might not hold in practice and labelled data is required in order to train a classifier. Without labels, transfer learning is possible to datasets under limited conditions (i.e., the source and target distributions should not be too dissimilar). In [26], two metrics are introduced to evaluate a single model playing against past and future versions of itself, as well as to measure the aptitude of two different fully trained models. In some way, this can be seen as an approximation of the minimax value we advocate in this paper, where instead of doing a full-on optimization in order to find the best adversary for the fixed generator, the search space is limited to discriminators that are snapshots from training, or discriminators trained with different seeds.

The ideas of duality and equilibria developed in the seminal work of [33, 22] have become a cornerstone in many fields of science, but are relatively unexplored for GANs. Some exceptions are [5, 10, 8, 13] but these works do not address the problem of evaluation. Closer to us, game theoretic metrics were previously mentioned in [25], but without a discussion addressing the stochastic nature and other practical difficulties of GANs, thus not yielding a practical applicable method. We conclude our discussion by pointing out the vast literature on duality used in the optimization community as a convergence criterion for min-max saddle point problems, see e.g. [23, 14]. Some recent work uses Lagrangian duality in order to derive an objective to train GANs [4] or to dualize the discriminator, therefore reformulating the saddle point objective as a maximization problem [17]. A similar approach proposed by [7] uses the dual formulation of Wasserstein GANs to train the decoder. Although we also make use of duality, there are significant differences. Unlike prior work, our contribution does not relate to *optimising GANs*. Instead, we focus on establishing that the duality gap acts as a proxy to measure convergence, which we do theoretically (Th. 1) as well as empirically, the latter requiring a new efficient estimation procedure discussed in Sec. 3.

2 Duality Gap as Performance Measure

Standard learning tasks are often described as (stochastic) optimization problems; this applies to common Deep Learning scenarios as well as to classical tasks such as logistic and linear regression. This formulation gives rise to a natural performance measure, namely the test loss[3]. In contrast, GANs are formulated as (stochastic) zero-sum games. Unfortunately, this fundamentally different formulation does not allow us to use the same performance metric. In this section, we describe a performance measure for GANs, which naturally arises from a game theoretic perspective. We start with a brief overview of zero-sum games, including a description of the *Duality gap* metric.

A zero-sum game is defined by two players $\mathcal{P}_1$ and $\mathcal{P}_2$ who choose a decision from their respective decision sets $\mathcal{K}_1$ and $\mathcal{K}_2$. A game objective $M : \mathcal{K}_1 \times \mathcal{K}_2 \mapsto \mathbb{R}$ sets the utilities of the players. Concretely, upon choosing a pure strategy $(\mathbf{u}, \mathbf{v}) \in \mathcal{K}_1 \times \mathcal{K}_2$ the utility of $\mathcal{P}_1$ is $-M(\mathbf{u}, \mathbf{v})$, while the utility of $\mathcal{P}_2$ is $M(\mathbf{u}, \mathbf{v})$. The goal of either $\mathcal{P}_1$/$\mathcal{P}_2$ is to maximize their worst case utilities:

$$\min_{\mathbf{u}\in\mathcal{K}_1} \max_{\mathbf{v}\in\mathcal{K}_2} M(\mathbf{u}, \mathbf{v}) \quad \textbf{(Goal of } \mathcal{P}_1\textbf{)}, \qquad \max_{\mathbf{v}\in\mathcal{K}_2} \min_{\mathbf{u}\in\mathcal{K}_1} M(\mathbf{u}, \mathbf{v}) \quad \textbf{(Goal of } \mathcal{P}_2\textbf{)} \tag{2}$$

This formulation raises the question of whether there exists a solution $(\mathbf{u}^*, \mathbf{v}^*)$ to which both players may jointly converge. The latter only occurs if there exists $(\mathbf{u}^*, \mathbf{v}^*)$ such that neither $\mathcal{P}_1$ nor $\mathcal{P}_2$ may

[3] For classification tasks using the zero-one test error is also very natural. Nevertheless, in regression tasks the test loss is often the only reasonable performance measure.

increase their utility by unilateral deviation. Such a solution is called a *pure equilibrium*, formally,

$$\max_{\mathbf{v}\in\mathcal{K}_2} M(\mathbf{u}^*,\mathbf{v}) = \min_{\mathbf{u}\in\mathcal{K}_1} M(\mathbf{u},\mathbf{v}^*) \quad \textbf{(Pure Equilibrium).}$$

While a pure equilibrium does not always exist, the seminal work of [22] shows that an extended notion of equilibrium always does. Specifically, there always exists a distribution $\mathcal{D}_1$ over elements of $\mathcal{K}_1$, and a distribution $\mathcal{D}_2$ over elements of $\mathcal{K}_2$, such that the following holds,

$$\max_{\mathbf{v}\in\mathcal{K}_2} \mathbb{E}_{\mathbf{u}\sim\mathcal{D}_1} M(\mathbf{u},\mathbf{v}) = \min_{\mathbf{u}\in\mathcal{K}_1} \mathbb{E}_{\mathbf{v}\sim\mathcal{D}_2} M(\mathbf{u},\mathbf{v}) \quad \textbf{(MNE).}$$

Such a solution is called a *Mixed Nash Equilibrium (MNE)*. This notion of equilibrium gives rise to the following natural performance measure of a given pure/mixed strategy.

Definition 1 (Duality Gap)**.** *Let $\mathcal{D}_1$ and $\mathcal{D}_2$ be fixed distributions over elements from $\mathcal{K}_1$ and $\mathcal{K}_2$ respectively. Then the duality gap DG of $(\mathcal{D}_1,\mathcal{D}_2)$ is defined as follows,*

$$\mathrm{DG}(\mathcal{D}_1,\mathcal{D}_2) := \max_{\mathbf{v}\in\mathcal{K}_2} \mathbb{E}_{\mathbf{u}\sim\mathcal{D}_1} \mathrm{M}(\mathbf{u},\mathbf{v}) - \min_{\mathbf{u}\in\mathcal{K}_1} \mathbb{E}_{\mathbf{v}\sim\mathcal{D}_2} \mathrm{M}(\mathbf{u},\mathbf{v}). \tag{3}$$

Particularly, for a given pure strategy $(\mathbf{u},\mathbf{v}) \in \mathcal{K}_1 \times \mathcal{K}_2$ *we define,*

$$\mathrm{DG}(\mathbf{u},\mathbf{v}) := \max_{\mathbf{v}'\in\mathcal{K}_2} \mathrm{M}(\mathbf{u},\mathbf{v}') \;-\; \min_{\mathbf{u}'\in\mathcal{K}_1} \mathrm{M}(\mathbf{u}',\mathbf{v})\,. \tag{4}$$

Two well-known properties of the duality gap are that it is always non-negative and is exactly zero in (mixed) Nash Equilibria. These properties are very appealing from a practical point of view, since it means that the duality gap gives us an immediate handle for measuring convergence.

Next we illustrate the usefulness of the duality gap metric by analyzing the ideal case where both G and D have unbounded capacity. The latter notion introduced by [9] means that the generator can represent any distribution, and the discriminator can represent any decision rule. The next proposition shows that in this case, as long as G is not equal to the true distribution then the duality gap is always positive. In particular, we show that the duality gap is at least as large as the Jensen-Shannon divergence between true and fake distributions. We also show that if G outputs the true distribution, then there exists a discriminator such that the duality gap (DG) is zero. See a proof in the Appendix.

Theorem 1 (DG and JSD)**.** *Consider the GAN objective in Eq. [1], and assume that the generator and discriminator networks have unbounded capacity. Then the duality gap of a given fixed solution $(G_{\mathbf{u}}, D_{\mathbf{v}})$ is lower bounded by the Jensen-Shannon divergence between the true distribution p_{data} and the fake distribution $q_{\mathbf{u}}$ generated by $G_{\mathbf{u}}$, i.e.* $\mathrm{DG}(\mathbf{u},\mathbf{v}) \geq \mathrm{JSD}(\mathrm{p_{data}}\,||\,\mathrm{q}_{\mathbf{u}})$. *Moreover, if $G_{\mathbf{u}}$ outputs the true distribution, then there exists a discriminator $D_{\mathbf{v}}$ such that* $\mathrm{DG}(\mathrm{G}_{\mathbf{u}},\mathrm{D}_{\mathbf{v}}) = 0$.

Note that different GAN objectives are known to be related to other types of divergences [24], and we believe that the Theorem above can be generalized to other GAN objectives [2, 11].

3 Estimating the Duality Gap for GANs

Appropriately estimating the duality gap from samples. Supervised learning problems are often formulated as stochastic optimization programs, meaning that we may only access estimates of the expected loss by using samples. One typically splits the data into training and test sets [4]. The training set is used to find a solution whose quality is estimated using a separate test set (which provides an unbiased estimate of the true expected loss). Similarly, GANs are formulated as stochastic zero-sum games (Eq. [1]) but the issue of evaluating the duality gap metric is more delicate. This is because we have three phases in the evaluation: **(i)** training a model $(\mathbf{u},\mathbf{v})$, **(ii)** finding the worst case discriminator/generator, $\mathbf{v}_{\text{worst}} \leftarrow \arg\max_{\mathbf{v}\in\mathcal{K}_2} M(\mathbf{u},\mathbf{v})$, and $\mathbf{u}_{\text{worst}} \leftarrow \arg\min_{\mathbf{u}\in\mathcal{K}_1} M(\mathbf{u},\mathbf{v})$, and **(iii)** computing the duality gap by estimating: $\mathrm{DG} := \mathrm{M}(\mathbf{u},\mathbf{v}_{\text{worst}}) - \mathrm{M}(\mathbf{u}_{\text{worst}},\mathbf{v})$. Now since we do not have direct access to the expected objective, one should use different samples for each of the three mentioned phases in order to maintain an unbiased estimate of the expected duality gap. Thus we split our dataset into three disjoint subsets: a training set, *an adversary finding set*, and a test set which are respectively used in phases **(i)**, **(ii)** and **(iii)**.

[4] Of course, one should also use a validation set, but this is less important for our discussion here.

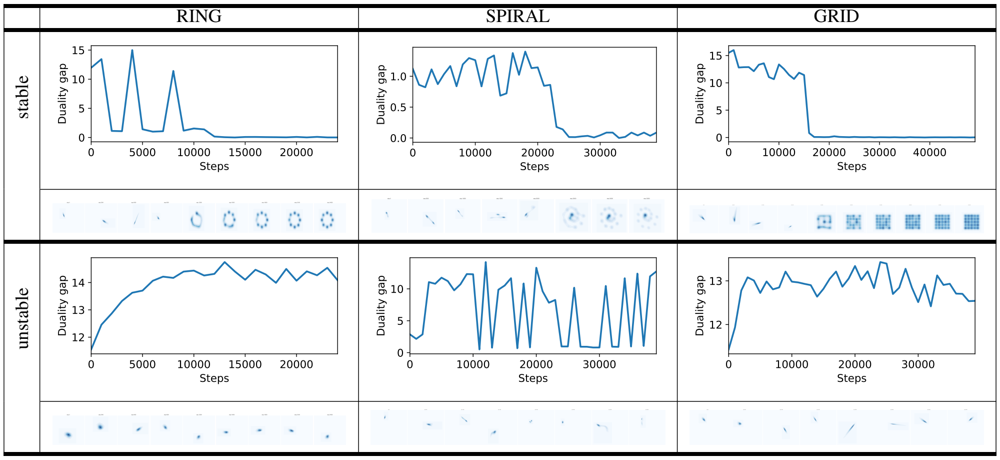


Figure 2: Progression of duality gap (DG) throughout training and heatmaps of generated samples.

Minimax Loss as a metric for evaluating generators. For all experiments, we report both the duality gap (DG) and the minimax loss $M(\mathbf{u}, \mathbf{v}_{\text{worst}})$. The latter is the first term in the expression of the DG and intuitively measures the 'goodness' of a generator $G_{\mathbf{u}}$. If $G_{\mathbf{u}}$ is optimal and covers p_{data}, the minimax loss achieves its optimal value as well. This happens when $D_{\mathbf{v}_{\text{worst}}}$ outputs 0.5 for both the real and generated samples. Whenever the generated distribution does not cover the entire support of p_{data} or compromises the sample quality, this is detected by $D_{\mathbf{v}_{\text{worst}}}$ and hence, the minimax loss increases. This makes it a compelling metric for detecting mode collapse and evaluating sample quality. Note that in order to compute this metric one only needs a batch of generated samples, i.e. the generator can be used as a black-box. Hence, this metric is not limited to generators trained as part of a GAN, but can instead be used for any generator that can be sampled from.

Practical and efficient estimation of the duality gap for GANs. In practice, the metrics are computed by optimizing a separate generator/discriminator using a gradient based algorithm. To speed up the optimization, we initialize the networks using the parameters of the adversary at the step being evaluated. Hence, if we are evaluating the GAN at step t, we train $\mathbf{v}_{\text{worst}}$ for $\mathbf{u}_t$ and $\mathbf{u}_{\text{worst}}$ for $\mathbf{v}_t$ by using $\mathbf{v}_t$ as a starting point for $\mathbf{v}_{\text{worst}}$ and analogously, $\mathbf{u}_t$ as a starting point for $\mathbf{u}_{\text{worst}}$ for a number of fixed steps. We also explored further approximations of DG, where instead of using optimization to find $\mathbf{v}_{\text{worst}}$ and $\mathbf{u}_{\text{worst}}$, we limit the search space to a set of discriminators and generators stored as snapshots throughout the training, similarly to [26] (see results in Appendix C.5). In Appendix D we include an in-depth analysis of the quality of the approximation of the DG and how it compares to the true theoretical DG.

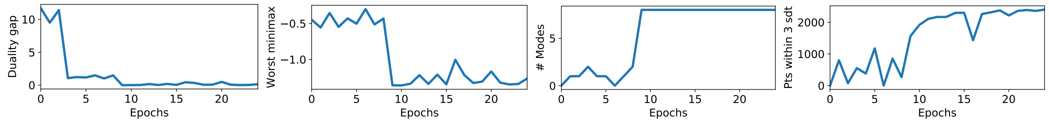
Non-negativity DG. While DG is non-negative in theory, this might not hold in practice since we only find approximate $(\mathbf{u}_{\text{worst}}, \mathbf{v}_{\text{worst}})$. Nevertheless, the practical scheme that we describe above makes sure that we do not get negative values for DG in practice. We elaborate on this in AppendixB.

4 Experimental results

We carefully design a series of experiments to examine commonly encountered failure modes of GANs and analyze how this is reflected by the two metrics. Specifically, we show the sensitivity of the duality gap metric to (non-) convergence and the susceptibility of the minimax loss to reflect the sample quality. Further details and additional extensive experiments can be found in the Appendix. Note that our goal is *not* to provide a rigorous comparative analysis between different GAN variants, but to show that both metrics capture properties that are particularly useful to monitor training.

4.1 Mixture of Gaussians

We train a vanilla GAN on three toy datasets with increasing difficulty: a) RING: a mixture of 8 Gaussians, b) SPIRAL: a mixture of 20 Gaussians and c) GRID: a mixture of 25 Gaussians. As the true data distribution is known, this setting allows for tracking convergence.



(a) DG (b) Minimax (c) Modes (d) Std

Figure 3: DG, minimax, modes covered and std. Tab. 7 in App. shows Pearson correlation between the metrics.

Duality gap and convergence. Our first goal is to illustrate the relation between convergence and the duality gap. To that end, we analyze the progression of DG throughout training in stable and unstable settings. One common problem of GANs is *unstable mode collapse*, where the generator alternates between generating different modes. We simulate such instabilities and compare them against successful GANs in Fig. 2. The gap goes to zero for all stable models after convergence to the true data distribution. Conversely, unstable training is reflected both in terms of the large value reached by DG as well as its trend over iterations (e.g., large oscillations and an increasing trend indicate unstable behavior). Thus the duality gap is a powerful tool for *monitoring the training and detecting unstable collapse*.

Minimax loss reflects sample quality. As previously argued, another useful metric to look at is the minimax loss which focuses solely on the generator. For the toy datasets, we measure the sample quality using *(i)* the number of covered modes and *(ii)* the number of generated samples that fall within 3 standard deviations of the modes (std). Fig. 3 shows significant anti-correlation, which indicates *the minimax loss can be used as a proxy for determining the overall sample quality*.

4.2 Duality gap and stable mode collapse

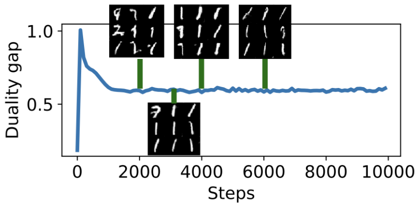


Figure 4: DG evolution detects stable mode collapse.

The previous experiment shows how *unstable* mode collapse is captured by DG - the trend is unstable and is typically within a high range. We are now interested in the case of *stable mode collapse*, where the model does converge, but only to a subset of the modes.

We train a GAN on MNIST where the generator collapses to generating from only one class and does not change the mode as the number of training steps increases. Fig. 4 shows the DG curve. The trend is flat and stable, but the value of DG is not zero, thus showing that *looking at both the trend and value of the DG is helpful for detecting stable mode collapse as well.*

4.3 The trend of the duality gap progression curve

We now analyze the trend of the DG curves. The plots (see Fig. 2) show it does not always monotonically decrease throughout training as we do observe non-smooth spikes. This raises the question as to whether these spikes are the result of the instabilities of the training or due to the metric itself? To address this, we train a GAN on a 2D submanifold Gaussian mixture embedded in 3D space. Such a setting captures a commonly encountered GAN failure as this mixture is degenerate with respect to the base measure defined in ambient space due to the lack of fully dimensional support.

It has been shown [29] that an unregularized GAN collapses in every run after 50K iterations (see Fig. 5 - right) because of the focus of the discriminator on smaller differences between the true and generated samples, whereas the training of a regularized version can essentially avoid collapsing even well beyond 200K iterations (see Fig. 5 - left). Thus we would expect the DG curves for the two settings to show different trends, which is indeed what we observe. For the unregularized version, DG decreases to small values and is stable until 40K steps, after which it starts increasing, reflecting the collapse of the generator. A practitioner looking at the DG curve can thus learn that (i) the training should be stopped between 20-40K steps and (ii) there is a collapse in the training (information that is especially valuable when the generated data is of non-image type or cannot be visualised). Conversely, the DG trend for the regularized version is stable and very quickly converges to values

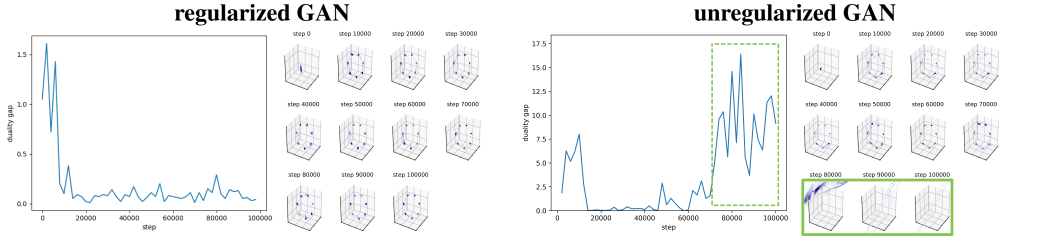


Figure 5: DG progression for a regularized (left) and unregularized GAN (right). Generated samples are shown for various steps. DG is able to capture instabilites (see green box).

close to zero, which reflects the improved quality. This suggests that *the usage of DG opens avenues to further understand and compare the effects of various regularizers*. Note that in [29] different levels of the regularizer were only visually compared due to the lack of a proper metric.

4.4 Comparison with image-specific criteria

We further analyze the sensitivity of the minimax loss to various changes in the sample quality for natural images that fall broadly in two categories: *(i)* mode sensitivity and *(ii)* visual sample quality. We compare against the commonly used Inception Score (INC) and Frechet Inception Distance (FID). Both metrics use the generator as a black-box through sampling. We follow the same setup for the evaluation of minimax and use the GAN zero-sum objective. Note that changing the objective to WGAN formulation makes it closely related to the Wasserstein critic [2].

Sensitivity to modes. As natural images are inherently multimodal, the generated distribution commonly ignores some of the true modes, which is a phenomenon known as *mode dropping*. Another common issue is *intra-mode collapse* that occurs when the generator is generating from all modes, but there is no variety within a mode. We then turn to *mode invention* where the generator creates non-existent modes. Fig. 18 in the Appendix shows the trends of all metrics for various degrees of mode dropping, invention and intra-mode collapse, where a class label is considered a mode. INC is unable to detect both intra-mode collapse and invented modes. On the other hand, *both FID and minimax loss exhibit desirable sensitivity to various mode changing.*

Sample quality. We study the metrics' ability to detect compromised sample quality by distorting real images using Gaussian noise, blur and swirl at an increasing intensity. As shown in Fig. 19 in the Appendix, *all metrics, including minimax, detect different degrees of visual sample quality.*

In Appendix C.3.1 we also show that the metric is *computationally efficient to be used in practice*. Tab. 1 gives a summary of all the results.

Poperty\Metric	**INC**	**FID**	**minimax**
Sensitivity to mode collapse	moderate	high	high
Sensitivity to mode invention	low	high	high
Sensitivity to intra-mode collapse	low	high	high
Sensitivity to visual quality and transformations	moderate	high	high
Computational: Fast	yes	yes	yes
Computational: Needs labeled data or a pretrained classifier	yes	yes	no
Computational: Can be applied to any domain without change	no	no	yes

Table 1: Comparison of INC, FID and minimax on various properties.

4.5 DG as a measure for the image domain

The previous section illustrates the metrics have desirable properties that makes them effective in capturing different failure modes in terms of image-specific criteria. Since generating images is one of the most common use cases of GANs, we further explore the usefulness of the DG on generating faces through a ProgGAN trained on CelebA. Figure 6 shows that unlike the GAN losses, the DG trend can capture the progress, which is also in agreement with the trend of the largest singular values of the convolutional layers of G and D.

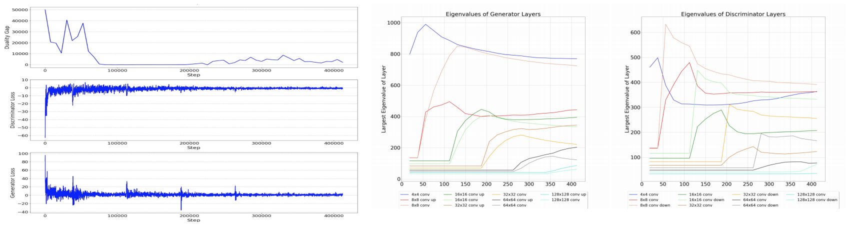


Figure 6: ProgGAN trained on CelebA: (left) losses vs. DG; (right) largest singular values of the conv layers

4.6 Generalization to other domains and GAN losses

These experiments test the ability of the two metrics to adapt to a different GAN loss formulation (WGAN-GP [11] and SeqGAN [34]), as well as other domains (cosmology, audio, text).

N-body simulations in cosmology. We consider the field of observational cosmology that relies on computationally expensive simulations with very different statistics from natural images. In an attempt to reduce this burden [28] trained a WGAN-GP to replace the traditional N-body simulators, relying on three statistics to assess the quality of the generated samples: mass histrogram, peak count and power spectral density. A random selection of real and generated samples shown in Fig. 21 in the Appendix demonstrate the high visual quality achieved by the generator.

We evaluate the agreement between the statistics of the real and generated samples using the squared norm of the statistics differences (lower scores are therefore better). In Fig. 7, we show the evolution of the scores corresponding to the three statistics as well as DG. We observe a strong correlation, especially between the peaks. Furthermore, it seems that the duality gap takes all the statistics into account. In Tab. 2, we observe a strong empirical correlation between the duality gap, the minimax value and the cosmological scores. We also observe that the FID is significantly less correlated that the duality gap, which is explained by the fact that the images we use are not natural images and hence that the statistics of the Inception Network are not suitable to evaluate the quality of cosmological samples.

	Mass hist.	Peak hist.	PSD
Dual gap	0.53	0.38	0.71
Minmax value	0.66	0.51	0.75
FID	0.40	0.21	0.34

Table 2: Pearson correlation between cosmological scores (mass histogram, peak histogram and Power Spectral Density (PSD)) and metrics (dual gap, minimax and FID).

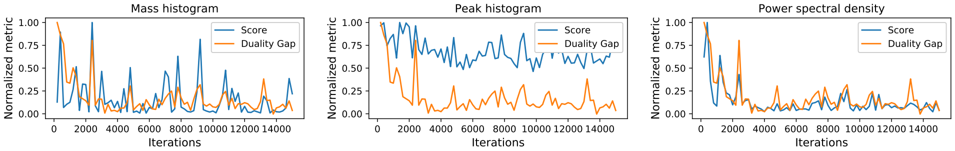


Figure 7: DG and cosmo-score evolution. DG strongly correlates with all 3 scores.

Audio Time-Frequency consistency. Generating an audio waveform is a challenging problem as it requires an agreement between scales from the range of milliseconds to tens of seconds. To overcome this challenge, one may use more powerful and intuitive features such as a Time-Frequency (TF) representation. Nevertheless, in order to obtain a natural listening experience, a TF representation needs to be consistent, i.e., there must exist an audio signal which leads to this TF representation. [19] define a measure estimating the consistency of a representation and use it to assess the convergence of their GAN. In Fig. 8, we show the evolution the measure and of DG and minimax. We observe a clear correlation, especially with the minimax value, which is expected as both the consistency measure and minimax evaluate only the generated samples.

Text generation. Another challenging modality for generative models is text. SeqGAN [34] is a GAN-based sequence framework evaluated by negative log-likelihood: *nll-oracle* and *nll-test*. The first metric computes the likelihood of generated data against an oracle, whereas the second makes use of the generator and the likelihood of real test data. Fig. 9 shows DG and minimax correlate well with both metrics. Moreover, both DG and minimax can determine the optimal stopping point.

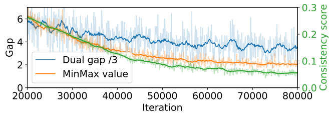


Figure 8: Evolution of the consistency measure vs. the DG.

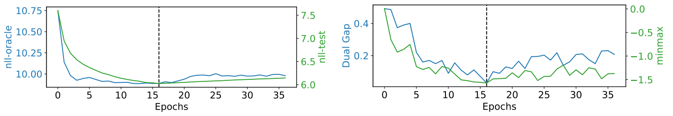


Figure 9: Evolution of nll-oracle and nll-test (left) vs. DG and minimax (right). The dashed line represents the optimal stopping point.

4.7 Comparison of different models

So far, we have seen that our estimation of DG generalizes to various loss functions and data distributions. We now turn our attention to evaluating the generalization abilities to models computed from different classes (i.e. neural networks with different architectures). We propose to do this evaluation by taking the original GAN minmax objective as a reference. The reasoning behind this choice is that a better discriminator would always be fooled less by a worst generator, irrespective of how it was trained. The analogue holds for the generator as well, resulting in a lower DG/minimax value on a selected objective.

	FID/DG	FID/mm	INC/DG	INC/mm
NS	0.32	0.88	-0.34	-0.87
NS + SN	0.93	0.85	-0.93	-0.91
NS + SN + GP	0.94	0.94	-0.91	-0.95
WGAN + GP	0.87	0.90	-0.87	0.91

Table 3: Pearson correlation between FID and INC with DG and minimax (mm) for different GAN variants.

To that end, we compare different commonly used ResNet based GAN variants on Cifar10: a GAN using the non saturating update rule (NS), spectrally normalized GAN (NS + SN), spectrally normalized GAN with the addition of a gradient penalty (NS + SN + GP) and a WGAN with gradient penalty (WGAN GP). We use the optimal hyperparameters suggested in [16]. Table 3 shows the Pearson correlation between DG and the minimax metric against FID and the Inception score (INC), which are known to work well on this dataset. We find that the minimax metric always highly correlates with both FID and INC. This is expected as (i) minimax evaluates the generator only, just as FID and INC that do not take the discriminator into consideration and (ii) as previously shown minimax is sensitive to mode changes and sample quality. Interestingly, DG also correlates highly whenever the discriminator is properly regularized. The level of correlation is however reduced for the unregularized variants. We hypothesise this is due to instabilities in the training, for which the generator might be improving, while the discriminator is becoming worse.

	FID		INC		DG		mm	
	score	rank	score	rank	score	rank	score	rank
NS	28.65	3	7.1	3	2.36	3	-1.07	3
NS+SN	22.25	1	7.78	1	0.21	1	-1.31	1
NS+SN+GP	23.46	2	7.75	2	0.23	2	-1.30	2
WGAN+GP	72.23	4	4.93	4	4.53	4	-0.2	4
compute time (*s*)	120.50		47.33		27.22		7.38	

Table 4: Scores and ranking for various GAN models using different metrics. The final row gives the computation time in seconds.

Tab. 4 shows the ranking of the models is the same for all four metrics, suggesting that *DG/minimax is a sensible choice for comparing different GAN models*.

5 Conclusion

We propose a domain agnostic evaluation measure for GANs that relies on the duality gap (DG) and upper bounds the JS divergence between real and generated data. This measure allows for meaningful monitoring of the progress made during training, which was lacking until now. We demonstrate that DG and its minimax part are able to detect various GAN failure modes (stable and unstable mode collapse, divergence, sample quality etc.) and rank different models successfully. Thus these metrics address two problems commonly faced by practitioners: 1) when should one stop training? and 2) if the training procedure has converged, have we reached the optimum? Finally, a significant advantage of the metrics is that, unlike many existing approaches, they require *no labelled data* and *no domain specific classifier*. Therefore, they are well-suited for applications of GANs other than the traditional generation task for images.

Acknowledgements. The authors would like to thank August DuMont Schütte for helping with the ProgGAN experiment and Gokula Krishnan Santhanam and Gary Bécigneul for useful discussions.

References

[1] A tensorflow implementation of "deep convolutional generative adversarial networks". https://github.com/carpedm20/DCGAN-tensorflow. Accessed: 2018-11-16.

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

[3] Ali Borji. Pros and cons of gan evaluation measures. *arXiv preprint arXiv:1802.03446*, 2018.

[4] Xu Chen, Jiang Wang, and Hao Ge. Training generative adversarial networks via primal-dual subgradient methods: A lagrangian perspective on gan. *arXiv preprint arXiv:1802.01765*, 2018.

[5] Constantinos Daskalakis, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. Training gans with optimism. *International Conference on Learning Representations (ICLR)*, 2018.

[6] Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Processing Magazine*, 29(6):141–142, 2012.

[7] Mevlana Gemici, Zeynep Akata, and Max Welling. Primal-dual wasserstein gan. *arXiv preprint arXiv:1805.09575*, 2018.

[8] Gauthier Gidel, Hugo Berard, Pascal Vincent, and Simon Lacoste-Julien. A variational inequality perspective on generative adversarial nets. *International Conference on Learning Representations (ICLR)*, 2019.

[9] 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.

[10] Paulina Grnarova, Kfir Y Levy, Aurelien Lucchi, Thomas Hofmann, and Andreas Krause. An online learning approach to generative adversarial networks. *International Conference on Learning Representations (ICLR)*, 2018.

[11] 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.

[12] 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.

[13] Ya-Ping Hsieh, Chen Liu, and Volkan Cevher. Finding mixed nash equilibria of generative adversarial networks. *arXiv preprint arXiv:1811.02002*, 2018.

[14] Nikos Komodakis and Jean-Christophe Pesquet. Playing with duality: An overview of recent primal? dual approaches for solving large-scale optimization problems. *IEEE Signal Processing Magazine*, 32(6):31–54, 2015.

[15] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. The cifar-10 dataset. *online: http://www. cs. toronto. edu/kriz/cifar. html*, 2014.

[16] Karol Kurach, Mario Lucic, Xiaohua Zhai, Marcin Michalski, and Sylvain Gelly. The gan landscape: Losses, architectures, regularization, and normalization. *arXiv preprint arXiv:1807.04720*, 2018.

[17] Yujia Li, Alexander Schwing, Kuan-Chieh Wang, and Richard Zemel. Dualing gans. In *Advances in Neural Information Processing Systems*, pages 5606–5616, 2017.

[18] David Lopez-Paz and Maxime Oquab. Revisiting classifier two-sample tests. *arXiv preprint arXiv:1610.06545*, 2016.

[19] Andrés Marafioti, Nicki Holighaus, Nathanaël Perraudin, and Piotr Majdak. Adversarial generation of time-frequency features with application in audio synthesis. *arXiv preprint arXiv:1902.04072*, 2019.

[20] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *International Conference on Machine Learning*, pages 3478–3487, 2018.

[21] Luke Metz, Ben Poole, David Pfau, and Jascha Sohl-Dickstein. Unrolled generative adversarial networks. *arXiv preprint arXiv:1611.02163*, 2016.

[22] John F Nash et al. Equilibrium points in n-person games. *Proceedings of the national academy of sciences*, 36(1):48–49, 1950.

[23] Arkadi Nemirovski, Anatoli Juditsky, Guanghui Lan, and Alexander Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.

[24] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 271–279. 2016.

[25] Frans A Oliehoek, Rahul Savani, Jose Gallego, Elise van der Pol, and Roderich Groß. Beyond local nash equilibria for adversarial networks. *arXiv preprint arXiv:1806.07268*, 2018.

[26] Catherine Olsson, Surya Bhupatiraju, Tom Brown, Augustus Odena, and Ian Goodfellow. Skill rating for generative models. *arXiv preprint arXiv:1808.04888*, 2018.

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

[28] Andres C. Rodriguez, Tomasz Kacprzak, Aurelien Lucchi, Adam Amara, Raphael Sgier, Janis Fluri, Thomas Hofmann, and Alexandre Réfrégier. Fast cosmic web simulations with generative adversarial networks. *arXiv preprint arXiv:1801.09070*, 2018.

[29] Kevin Roth, Aurelien Lucchi, Sebastian Nowozin, and Thomas Hofmann. Stabilizing training of generative adversarial networks through regularization. In *Advances in Neural Information Processing Systems*, pages 2018–2028, 2017.

[30] Mehdi SM Sajjadi, Olivier Bachem, Mario Lucic, Olivier Bousquet, and Sylvain Gelly. Assessing generative models via precision and recall. In *Advances in Neural Information Processing Systems*, pages 5228–5237, 2018.

[31] 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.

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

[33] J Von Neumann. Zur theorie der gesellschaftsspiele. *Mathematische annalen*, 100(1):295–320, 1928.

[34] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. Seqgan: Sequence generative adversarial nets with policy gradient. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[35] Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. Texygen: A benchmarking platform for text generation models. *SIGIR*, 2018.

A Proof of Theorem 1

Theorem 1 (DG and JSD)**.** *Consider the GAN objective in Eq. [1], and assume that the generator and discriminator networks have unbounded capacity. Then the duality gap of a given fixed solution* $(G_{\mathbf{u}}, D_{\mathbf{v}})$ *is lower bounded by the Jensen-Shannon divergence between the true distribution* $\mathrm{p_{data}}$ *and the fake distribution* $\mathrm{q_{\mathbf{u}}}$ *generated by* $G_{\mathbf{u}}$*, i.e.* $\mathrm{DG}(\mathbf{u}, \mathbf{v}) \geq \mathrm{JSD}(\mathrm{p_{data}} \,||\, \mathrm{q_{\mathbf{u}}})$*. Moreover, if* $G_{\mathbf{u}}$ *outputs the true distribution, then there exists a discriminator* $D_{\mathbf{v}}$ *such that* $\mathrm{DG(G_{\mathbf{u}}, D_{\mathbf{v}})} = 0$*.*

Proof. Let us denote by $p(x)$ the distribution over true samples and by $q_{\mathbf{u}}(x)$ the distribution over fake samples generated by the generator $G_{\mathbf{u}}$. Let us also denote the output of the discriminator by $D_{\mathbf{v}}(x)$. For simplicity, we will also slightly abuse notation and denote the GAN objective by $M(q_{\mathbf{u}}, D_{\mathbf{v}})$. Thus, the GAN objective reads as follows,

$$M(q_{\mathbf{u}}, D_{\mathbf{v}}) := \frac{1}{2}\int p(x) \log D_{\mathbf{v}}(x)dx + \frac{1}{2}\int q_{\mathbf{u}}(x)\log(1 - D_{\mathbf{v}}(x))dx \ . \tag{5}$$

First we prove the first part of the proposition: Let us first recall the definition of the Jensen-Shannon divergence of two distributions $p(\cdot), q_{\mathbf{u}}(\cdot)$,

$$\mathrm{JSD(p \,||\, q_{\mathbf{u}})} := \frac{1}{2}\mathrm{KL}\left(\mathrm{p} \,||\, \frac{\mathrm{p + q_{\mathbf{u}}}}{2}\right) + \frac{1}{2}\mathrm{KL}\left(\mathrm{q_{\mathbf{u}}} \,||\, \frac{\mathrm{p + q_{\mathbf{u}}}}{2}\right) \ . \tag{6}$$

where the KL divergence is defined as,

$$\mathrm{KL}(p \,||\, q_{\mathbf{u}}) := \int p(x) \log\left(\frac{p(x)}{q_{\mathbf{u}}(x)}\right) dx \ .$$

Now given a fixed solution $(q_{\mathbf{u}}, D_{\mathbf{v}})$ we will show that the duality gap of this pair is bounded by the Jensen-Shannon divergence. It is well known that this divergence equals zero if both distributions are equal[5], and is otherwise strictly positive. To do so, we will first bound the minimax/maximin values for $q_{\mathbf{u}}/D_{\mathbf{v}}$.

(a) Upper Bounding Minimax Value: Given $q_{\mathbf{u}}(x)$, the worst case discriminator is obtained by taking the derivative of the objective in Equation (5) with respect to $D_{\mathbf{v}}(x)$ separately for every x (this can be done since we assume the capacity of $D_{\mathbf{v}}$ to be unbounded). This gives the following worst case discriminator (see similar derivation in [9]),

$$D_{\mathbf{v}}^{\max}(x) := \frac{p(x)}{p(x) + q_{\mathbf{u}}(x)} \ .$$

Plugging the above value into Equation (5) gives the following minimax value,

$$\begin{aligned}\max_{D_{\mathbf{v}}} M(q_{\mathbf{u}}, D_{\mathbf{v}}) &= M(q_{\mathbf{u}}, D_{\mathbf{v}}^{\max}) \\ &= \frac{1}{2}\int p(x)\log\left(\frac{p(x)}{q_{\mathbf{u}}(x) + p(x)}\right)dx + \frac{1}{2}\int q_{\mathbf{u}}(x)\log\left(\frac{q_{\mathbf{u}}(x)}{q_{\mathbf{u}}(x)+p(x)}\right)dx \\ &= -\log 2 + \mathrm{JSD(p \,||\, q_{\mathbf{u}})} \end{aligned} \tag{7}$$

(b) Lower Bounding Maximin Value: Here we lower bound the maximin value for a given $q_{\mathbf{u}}(x)$,

$$\min_{q_{\mathbf{u}}} M(q_{\mathbf{u}}, D_{\mathbf{v}}) \leq M(p, D_{\mathbf{v}}) = \frac{1}{2}\int p(x)\log D_{\mathbf{v}}(x)dx + \frac{1}{2}\int p(x)\log(1 - D_{\mathbf{v}}(x))dx \ . \tag{8}$$

Maximizing the last expression separately for every x gives

$$\max_{D_{\mathbf{v}}(x)\in[0,1]} \frac{1}{2}p(x)\log D_{\mathbf{v}}(x)dx + \frac{1}{2}p(x)\log(1 - D_{\mathbf{v}}(x))dx = -\log 2$$

Plugging the above into Equation (8) gives,

$$\min_{q_{\mathbf{u}}} M(q_{\mathbf{u}}, D_{\mathbf{v}}) \leq -\log 2 \ . \tag{9}$$

[5] We mean equal up to sets of measure zero.

(c) Upper bound on Duality Gap: Recall the definition of Duality gap,

$$\mathrm{DG}(\mathrm{q}_{\mathbf{u}}, \mathrm{D}_{\mathbf{v}}) := \max_{\mathrm{D}_{\mathbf{v}}} \mathrm{M}(\mathrm{q}_{\mathbf{u}}, \mathrm{D}_{\mathbf{v}}) - \min_{\mathrm{q}_{\mathbf{u}}} \mathrm{M}(\mathrm{q}_{\mathbf{u}}, \mathrm{D}_{\mathbf{v}}) \ .$$

Using Equation (7) together with Equation (9) immediately shows that

$$\mathrm{DG}(\mathrm{q}_{\mathbf{u}}, \mathrm{D}_{\mathbf{v}}) \geq \mathrm{JSD}(\mathrm{p} \,||\, \mathrm{q}_{\mathbf{u}}) \tag{10}$$

Therefore the duality gap is lower bounded by the Jensen-Shannon divergence between true and fake distributions, which concludes the first part of the proof.

Next we prove the second part of the proposition: Recall that we assume $q_{\mathbf{u}}(x) = p(x)$. And let us take,

$$D_{\mathbf{v}}(x) = \frac{1}{2}, \qquad \forall x$$

Next we show that the Duality gap of $(G_{\mathbf{u}}, D_{\mathbf{v}})$ is zero.

(a) Let us first compute the minimax value: Similarly to Equation (7) the following can be shown,

$$\begin{aligned} M(q_{\mathbf{u}}, D_{\mathbf{v}}^{\max}) &= \frac{1}{2}\int p(x)\log\left(\frac{p(x)}{q_{\mathbf{u}}(x)+p(x)}\right)dx + \frac{1}{2}\int q_{\mathbf{u}}(x)\log\left(\frac{q_{\mathbf{u}}(x)}{q_{\mathbf{u}}(x)+p(x)}\right)dx \\ &= -\log 2 \cdot \frac{1}{2}\int (q_{\mathbf{u}}(x) + p(x))dx \\ &= -\log 2 \ . \end{aligned} \tag{11}$$

where we used $p(x)/(p(x) + q_{\mathbf{u}}(x)) = \frac{1}{2}$.

(b) Let us now compute the maximin value. Since $D_{\mathbf{v}}(x) = \frac{1}{2}$ the following holds for any $q^0_{\mathbf{u}}(x)$,

$$M(q^0_{\mathbf{u}}, D_{\mathbf{v}}) = \frac{1}{2}\int p(x)\log D_{\mathbf{v}}(x)dx + \frac{1}{2}\int q^0_{\mathbf{u}}(x)\log(1 - D_{\mathbf{v}}(x))dx = -\log 2 \ ,$$

which immediately implies,

$$\min_{q_{\mathbf{u}}} M(q_{\mathbf{u}}, D_{\mathbf{v}}) = -\log 2 \ . \tag{12}$$

Combining Equation (11) with Equation (12), with the definition of the Duality gap implies,

$$\mathrm{DG}(\mathrm{q}_{\mathbf{u}}, \mathrm{D}_{\mathbf{v}}) = 0 \ .$$

which concludes the second part of the proof.

□

Approximate computation of the duality gap Note that the computation of the duality gap requires finding the exact solution for $\min_{q_{\mathbf{u}}} M(q_{\mathbf{u}}, D_{\mathbf{v}})$, and $\max_{D_{\mathbf{v}}} M(q_{\mathbf{u}}, D_{\mathbf{v}})$. In practice it is reasonable to assume that we may solve these two optimization problems only up to some approximation ε, i.e., that we may compute $q^{*,\varepsilon}_{\mathbf{u}}$ and $D^{*,\varepsilon}_{\mathbf{v}}$ such that,

$$M(q^{*,\varepsilon}_{\mathbf{u}}, D_{\mathbf{v}}) \leq \min_{q_{\mathbf{u}}} M(q_{\mathbf{u}}, D_{\mathbf{v}}) + \varepsilon, \qquad \& \qquad M(q_{\mathbf{u}}, D^{*,\varepsilon}_{\mathbf{v}}) \geq \max_{D_{\mathbf{v}}} M(q_{\mathbf{u}}, D_{\mathbf{v}}) - \varepsilon$$

In this case, a simple adaptation to the derivation above shows that the approximate duality gap computed using $q^{*,\varepsilon}_{\mathbf{u}}$ and $D^{*,\varepsilon}_{\mathbf{v}}$ gives us an upper bound on the approximate JS divergence as follows,

$$\mathrm{DG}_{\varepsilon}(\mathrm{q}_{\mathbf{u}}, \mathrm{D}_{\mathbf{v}}) := \mathrm{M}(\mathrm{q}_{\mathbf{u}}, \mathrm{D}^{*,\varepsilon}_{\mathbf{v}}) - \mathrm{M}(\mathrm{q}^{*,\varepsilon}_{\mathbf{u}}, \mathrm{D}_{\mathbf{v}}) \geq \mathrm{JSD}(\mathrm{p} \,||\, \mathrm{q}_{\mathbf{u}}) - 2\varepsilon \ .$$

B Non-negativity of the practical duality gap.

While the DG is guaranteed to be non-negative in theory, this might not hold in practice since we do not have access to the exact $\arg\min$ and $\arg\max$ but instead optimize for a fixed number of steps. Therefore, a question that arises is whether the non-negativity property is affected by practical approximations of DG. First, note that negative DG values occur if at some step t, $M(\mathbf{u}_t, \mathbf{v}_{\text{worst}}) < M(\mathbf{u}_{\text{worst}}, \mathbf{v}_t)$. However, the optimization algorithm yields $M(\mathbf{u}_t, \mathbf{v}_{\text{worst}}) > M(\mathbf{u}_t, \mathbf{v}_t)$ since $\mathbf{v}_{\text{worst}}$ is initialized using $\mathbf{v}_t$ and optimized to maximize the objective. Similarly, we expect $M(\mathbf{u}_t, \mathbf{v}_t) > M(\mathbf{u}_{\text{worst}}, \mathbf{v}_t)$, and DG is therefore non-negative. This of course assumes that the optimizer uses an appropriate set of parameters that guarantee successful decrease/increase of the objective. In Appendix D, we investigate the impact of the practical approximation of DG in lieu of the exact computation. In particular, we find that - both in theory and in practice - DG is not affected by the presence of mode collapse in $\mathbf{u}_{\text{worst}}$.

	stable		unstable	
	lr G	lr D	lr G	lr D
RING	1e-3	1e-4	1e-4	2e-4
SPIRAL	1e-3	2e-3	1e-4	2e-3
GRID	1e-3	2e-3	1e-4	2e-3

Table 5: Learning rates used for the toy experiments.

		Duality Gap	Modes	Std
stable	RING	0.04	8	2375
	SPIRAL	0.14	20	1999
	GRID	0.03	25	2370
unstable	RING	13	2	152
	SPIRAL	1.22	1	1724
	GRID	12.09	3	37

Table 6: Final results for DG, number of covered modes and number of generated samples (out of 2400) that fall within 3 standard deviations of the means.

C Experiments

C.1 Toy Dataset: Mixture of Gaussians

The toy datasets consist of a mixture of 8, 20 and 25 Gaussians for each of the models (RING, SPIRAL, GRID), respectfully. The standard deviation is set to 0.05 for all models except for the RING where the std is 0.01. Depending on the dataset, the means are spaced equally around a unit circle, a spiral or a grid.

The architecture of the generator consists of two fully connected layers (of size 128) and a linear projection to the dimensionality of the data (i.e. 2). The activation functions for the fully connected layers are relu. The discriminator is symmetric and hence, composed of two fully connected layers (of size 128) followed by a linear layer of size 1. The activation functions for the fully connected layers are relu, whereas the final layer uses sigmoid as an activation function.

Adam was used as an optimizer for both the discriminator and the generator with $beta_1 = 0.5$ and a batch size of 100. The latent dimensionality z is 100. The learning rates for the reported models are given as follows in Table 5. The optimizer used for training the worst D/G is Adam and is set to the default parameters.

Plots of DG during training are given in Table 2. Table 6 lists the obtained results for the methods in terms of their final duality gap, number of modes they have covered and the number of generated points that fall within three standard deviations of one of the means. The heatmaps of the final generated distributions are given in Figure 10.

We also plot generated samples from the worst case generator in Figure 11.

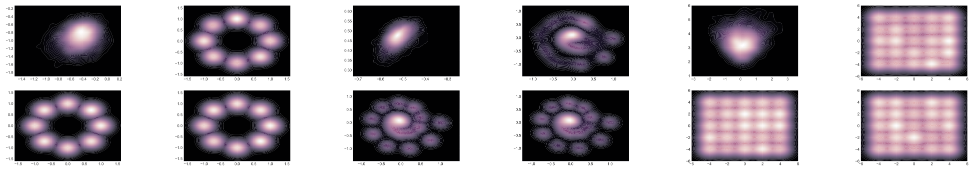
Progress during training for Figure 13 is given in Figure 12.

Finally, the progress of DG, minimax, number of modes, and generated samples close to modes across epochs is given in Figure 13. High correlation can be observed, which matches the quantitive results in Table 7.

The correlation values are reported in Table 7. We observe significant anti-correlation (especially for minimax loss), which indicates that both metrics capture changes in the number of modes and hence *the minimax loss can be used as a proxy to determining the overall sample quality*.

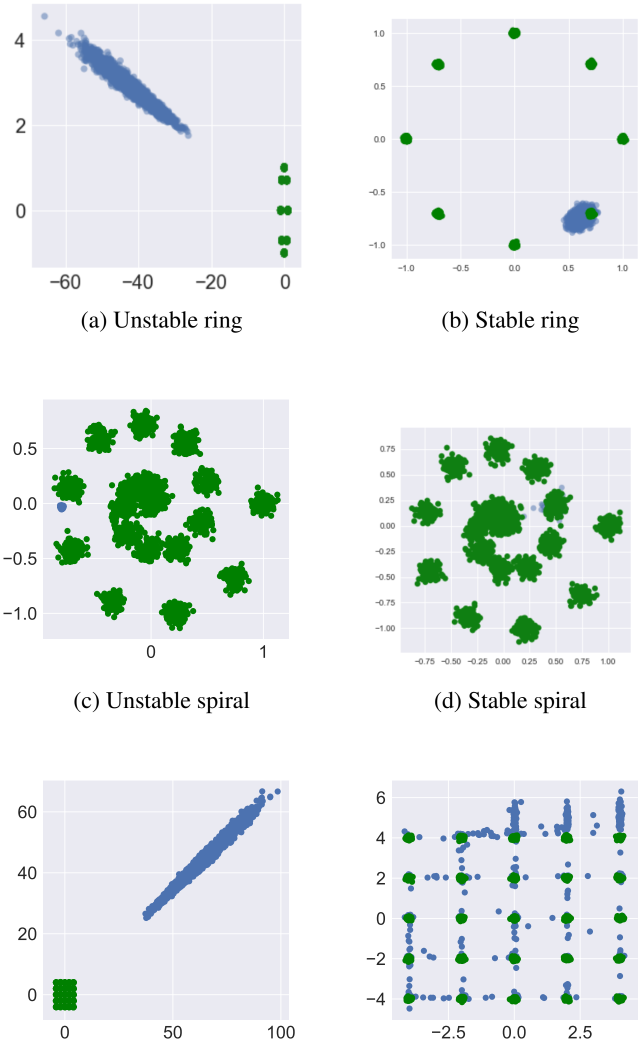
C.1.1 Loss Curves

A common problem practitioners face is when to stop training, i.e. understanding whether the model is still improving or not. See for example Figure 14, which shows the discriminator and generator losses during training of a DCGAN model on CIFAR10 [1]. The training curves are oscillating and



(a) Unstable ring (b) Stable ring (c) Unstable spiral (d) Stable spiral (e) Unstable grid (f) Stable grid

Figure 10: Heatmaps of the generated distributions at the final steps. On top: trained model (stable or unstable), on bottom: p_{data}



(a) Unstable ring

(b) Stable ring

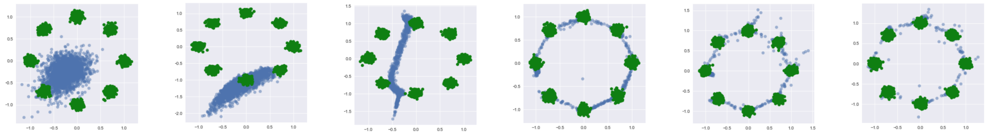
(c) Unstable spiral

(d) Stable spiral

(e) Unstable grid

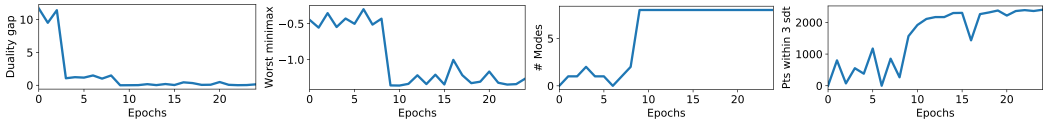
(f) Stable grid

Figure 11: Generated samples (in blue) from the worst generator for the discriminator for both the stable and unstable models. (ground truth in green color).



(a) Step0 (b) Step5 (c) Step10 (d) Step15 (e) Step20 (f) Step25

Figure 12: Generated samples (blue) and real samples (green) throughout training steps



(a) DG (b) Minimax (c) Modes (d) Std

Figure 13: DG, minimax, number of modes, and generated samples close to modes across epochs

	DG	Minimax	
Modes	-0.63	-0.97	ring
	-0.59	-0.93	spiral
	-0.71	-0.95	grid
Std	-0.64	-0.94	ring
	-0.64	-0.58	spiral
	-0.7	-0.93	grid

Table 7: Pearson product-moment correlation coefficients for an average of 10 stable rounds. Progress throughout the training of the individual metrics can be seen in Figure 13

hence are very non-intuitive. A practitioner needs to most often rely on visual inspection or some performance metric as a proxy as a stopping criteria.

The generator and discriminator losses for our 2D ring problem are shown in Figure 15. Based on the curves it is hard to determine when the model stops improving. As this is a 2D problem one can visually observe when the model has converged through the heatmaps of the generated samples (see Table 2). However in higher-dimensional problems (like the one discussed above on CIFAR10) one cannot do the same. Figure 16 showcases the progression of the duality gap throughout the training. Contrary to the discriminator/generator losses, this curve is meaningful and clearly shows the model has converged and when one can stop training, which coincides with what is shown on the heatmaps.

C.2 Other hyperparameters

C.2.1 Stable Mode Collapse

The architecture of the generator consists of 5 fully connected layers of size 128 with leaky relu as an activation unit, followed by a projection layer with tanh activation. The discriminator consists of 2 dense layers of size 128 and a projection layer. The activation function used for the dense layers of the discriminator is leaky relu as well, while the final layer uses a sigmoid. The value α for leaky relu is set to 0.3.

The optimizer we use is Adam with default parameters for both the generator, discriminator, as well as the optimizers for training the worst generator and discriminator. The dimensionality of the latent space z is set to 100 and we use a batch size of 100 as well. We train for 10K steps. The number of steps for training the worst case generator/discriminator is 400.

We use the training, validation and test split of MNIST [6] for training the GAN, training the worst case generator/discriminator, and estimating the duality gap (as discussed in Section 3).

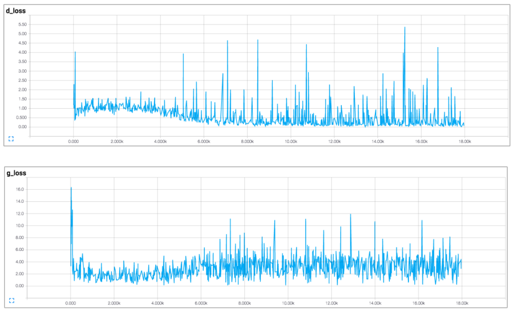
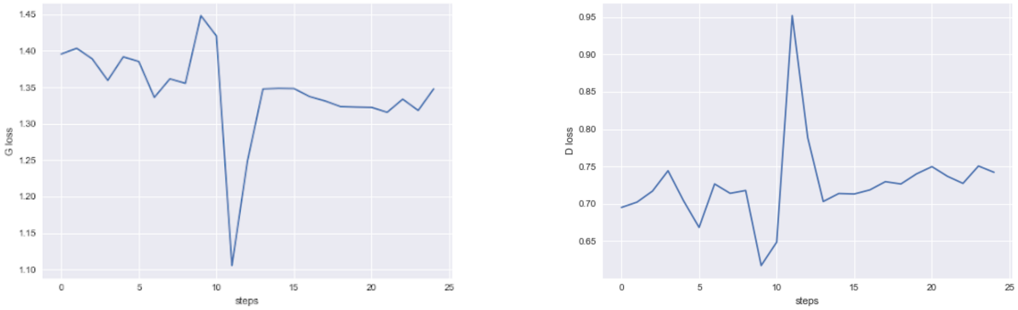


Figure 14: Discriminator and generator loss curves for a DCGAN model trained on CIFAR10. The curves are oscillating and it is hard to determine when to stop the training.



(a) Generator loss

(b) Discriminator loss

Figure 15: Discriminator and generator loss curves for the 2D ring problems. The curves are oscillating and it is hard to determine when to stop the training and when the model stops improving.

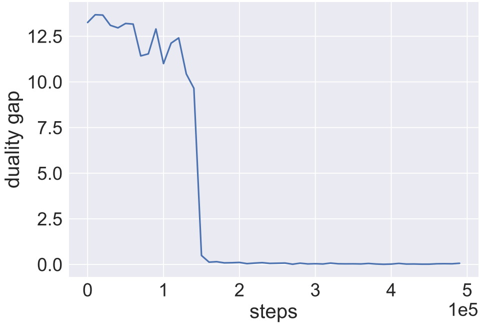


Figure 16: Curve of the progression of the duality gap during training.

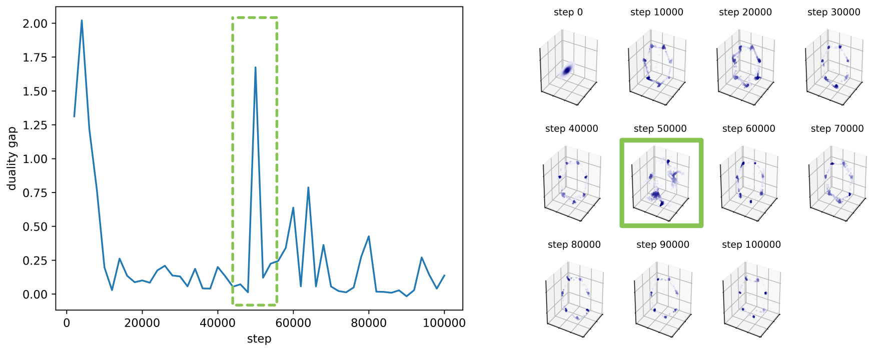


Figure 17: Spikes in the DG progression curve reflect the quality of the generated samples

C.3 3D GAN experiment

We train a simple GAN on a 2D submanifold mixture of seven Gaussians arranged in a circle and embedded in 3D space following the setup by [29]. The strength of the regularizer γ is set to 0.1, and respectively, to 0 for the unregularized version.

Figure 17 shows a duality gap progression curve for a regularized GAN. It can be observed that the the spikes that occur are due to the training properties, i.e. at the particular step when there is an apparent spike, the quality of the generated samples worsens.

C.3.1 Minimax experiment on Cifar10

Here we describe hyperaparameters for the experiment on Cifar10 [15]. The worst case discriminator we train is using the commonly used DCGAN architecture [27]. We again use Adam with the default parameters as the optimizer and the batch size is 100. We update the worst case classifier for 1K steps.

The hyperparameters used for the distortion in the experiment for visual sample quality are:

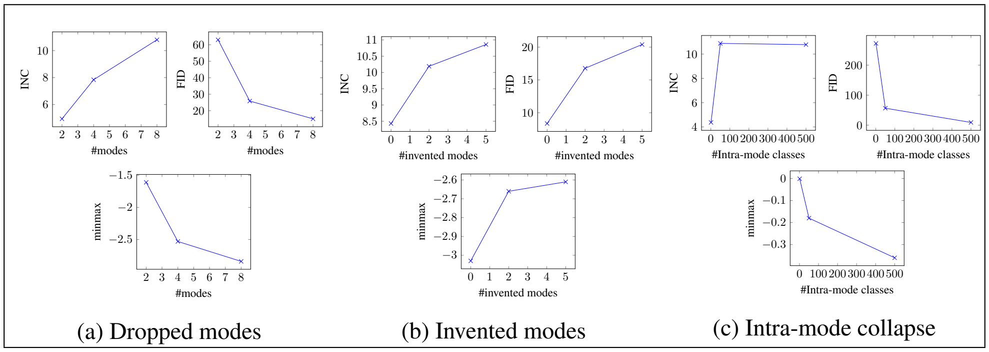
1. Gaussian noise
   (a) level 1: $sigma = 5$
   (b) level 2: $sigma = 10$
   (c) level 3: $sigma = 20$
2. Gaussian blur
   (a) level 1: $ksize = 2$
   (b) level 2: $ksize = 5$
   (c) level 3: $ksize = 7$
3. Gaussian swirl with strength 5
   (a) level 1: $radius = 1$
   (b) level 2: $radius = 2$
   (c) level 3: $radius = 20$

Mode sensitivity and sample quality detection. Fig. 18 shows the ability of the three metrics (INC, FID and minimax) to detect mode dropping, mode invention and intra-mode collapse. We simulate mode dropping by using the class labels as modes. The input is a set of 5K images containing all 10 classes as 'real' images, and another set of 5K images composed of only subset of the modes (subset of 2, 4 and 8) as 'generated' images (Figure 18 a).

We then turn to *mode invention* where the generator creates non-existent modes. For this setting, the set of 'real' images contains only 5 classes, whereas the sets of 'generated' images are supersets of 5, 7 and 10 classes (Figure 18 b). *Intra-mode collapse* is another common issue that occurs when the generator is generating from all modes, but there is no variety within a mode. The 'generated' sets consist of images from all 10 classes, but contain only 1, 50 and 500 unique images within a class.

Figure 18 shows the trends for all metrics for the various degrees of mode dropping, invention and intra-class collapse. INC is unable to detect both intra-mode collapse and invented modes. On the other hand, *both FID and minimax loss exhibit desirable sensitivity to various mode changing.*

The ability of the metrics to react to distortion at an increasing intensity is in Fig. 19.



(a) Dropped modes

(b) Invented modes

(c) Intra-mode collapse

Figure 18: INC, FID and minimax loss for a) mode collapse (x-axis: how many modes out of 10 are generated); b) mode invention (x-axis: how many invented modes are generated) and c) intra-mode collapse (x-axis: number of unique images within a class). For INC higher is better; for FID and minimax lower is better.

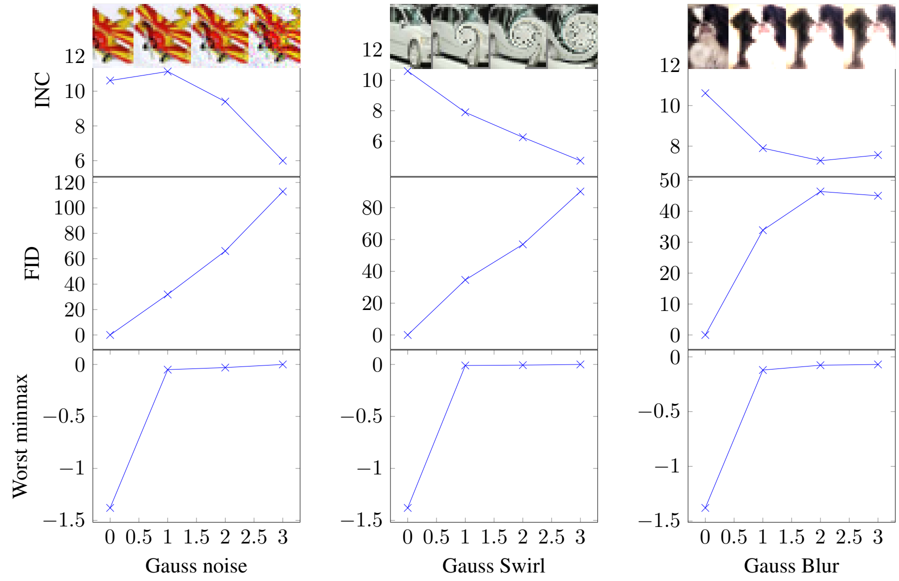


Figure 19: INC, FID and minimax loss on samples at increasing intensity of disturbance

Efficiency. A metric needs to be computationally efficient in order to be used in practice to both track the progress during training and as a final metric to rank various models. Figure 20 shows the wall clock time in terms of seconds for all three metrics. We keep the number of update steps fixed to 1K which makes the computation of the minimax loss efficient, as are the other two metrics. The computation of DG takes twice the time of the computation of the minimax loss.

We also test the variance across rounds due to randomness in the seed and how this affects the final metric and overall ranking, on a simple mode collapse task. Table 8 summarizes the average of 5 rounds showing that the variance is negligible and does not affect the effectiveness of the metric.

C.3.2 Experiment on cosmology dataset

Following the approach of [11], we used a Wasserstein loss with a gradient penalty of 10. Both the generator and the discriminator were optimized with an "RMSprop" optimizer and a learning rate of $3 \cdot 10^{-5}$. The discriminator was optimized 5 times more often than the generator.

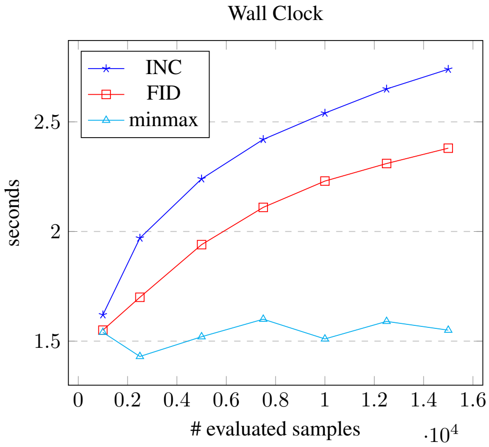


Figure 20: Wall clock time (in log-scale) for the calculation of INC, FID and minimax loss for increasing number of samples to be processed

classes	INC	FID	Minimax
2 classes	4.94±0.13	63	-1.69±0.06
4 classes	7.84±0.3	25.85	-2.53±0.04
6 classes	9.88±0.18	18.39	-3.14±0.03

Table 8: Metrics on a simple mode dropping task

Real and generated samples are given side-by-side in Figure 21, showcasing the quality of the trained model.

C.3.3 Experiment on text generation

For the implementation of SeqGAN, we used the TexyGen framework [35] and their preselected hyperparameters and metrics. Some generated samples are available in Tab9.

C.4 Experiment on CIFAR10 for comparison of different GAN variants

We follow the setup and best hyperparameters reported in [16]. In particular, we train every model for 200K training steps and evaluate using 10K samples. For the computation of DG and minimax, we use 150 training steps and a train/test split of 9600/400 samples.

Figures 22, 23 and 24 show samples generated from the worst generator at various steps throughout training.

C.5 Approximating the duality gap

We explored variants in which we are circumventing the optimization in order to find the worst case generator/discriminator by using a set of discriminators/generators out of which we choose the most adversarial one. The sets are created by saving snapshots of the parameters of the two networks during training. We explored variants where the snapshots come a) only from past models and b) a mix of previous and future models, and generally found b) to perform better. This setting is similar to the models proposed in [26], except they use skill rating systems to infer a latent variable for the

a man on his cell phone on ita cat is taking a picture of the door to the dark open sits .
a white bathroom with a black tub and a urinal and a toilet and a toilet .
a motorcycle parked on the sidewalk .
a motorcycle parked on a street looking up the street .
a living area is full wooden floor with plants growing fly to the toilet and a sinkairplane is , parked in a parking lot .

Table 9: Generated samples from SeqGAN.

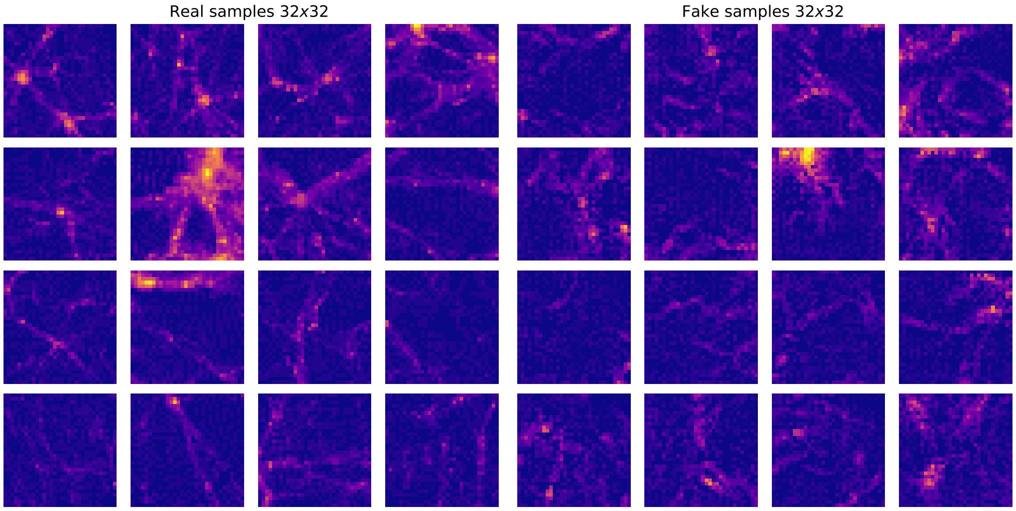


Figure 21: Real and generated cosmological images, representing slices of mass density of the universe. The yellow coefficient implies a high mass concentration

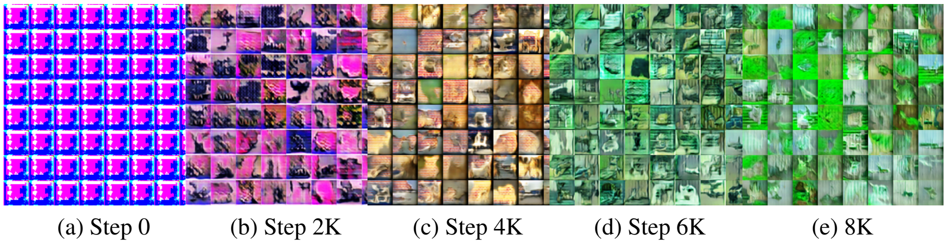

(a) Step 0 (b) Step 2K (c) Step 4K (d) Step 6K (e) 8K

Figure 22: a-e: Generated samples at different training steps from the worst generator

successfulness of a generator. On the other hand, we compute the duality gap and the minimax loss to infer the successfulness of the entire GAN and of the generator, respectfully.

Table 10 gives the progression of the approximated duality gap for four various scenarios: stable model, unstable mode collapse and stable mode collapse. The duality gap was approximated using 10 models that spanned accross 2 epochs.

D Analysis of the quality of the empirical DG

The theoretical assumption appearing in the proof in Appendix A is that the discriminator and generator have unbounded capacity and we can obtain the true minimizer and maximizer when computing $\mathbf{u}_{\text{worst}}$ and $\mathbf{v}_{\text{worst}}$, respectively. This, however, is not tractable in practice. Furthermore, it is well known that one common problem in GANs is mode collapse. This raises the question of how

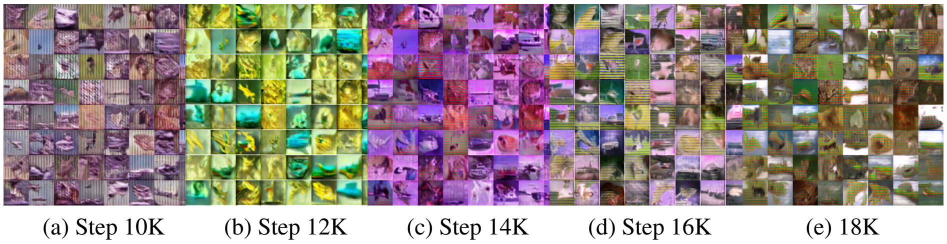
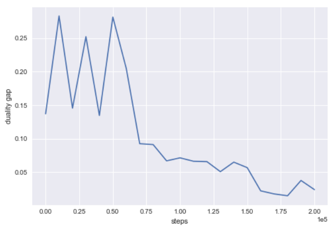
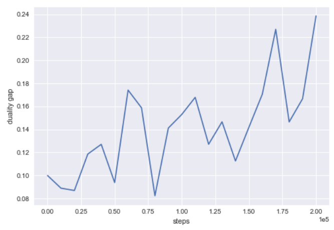
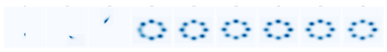
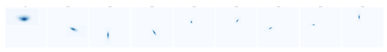
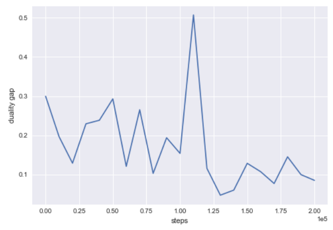
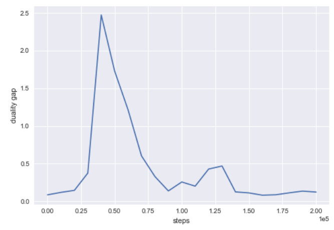
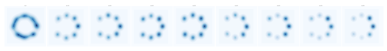
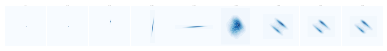

(a) Step 10K (b) Step 12K (c) Step 14K (d) Step 16K (e) 18K

Figure 23: a-e: Generated samples at different training steps from the worst generator

(a) Step 20K

Figure 24: a: Generated samples at the final step from the worst generator

Table 10: Progression of DG throughout training and heatmaps of the generator distribution. A1: stable ring, A2: unstable ring, B1: mode collapse, B2: stable mode collapse

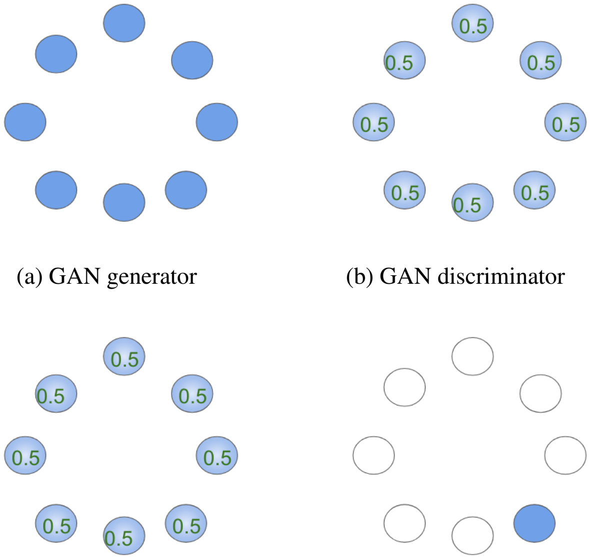
	1	2
A		
		
B		
		

the duality gap metric would be affected if the worst generator that we compute is collapsed itself. In the following we address this both empirically and from a theoretical perspective.

We use the same experimental setup as described in Appendix C.1. We focus on a GAN that has converged such that the generator covers all modes uniformly i.e. $p_g = p_{data}$ (Figure 25 a)). The discriminator outputs 0.5 for real and fake samples (Figure 25 b)). This means that the model has reached the equilibrium and the duality gap -in theory- is zero.

Collapsed worst case generator. Now we focus on the calculation of the duality gap. Let us consider the case of a mode collapsed worst case generator. In particular, when computing the maximin part of the duality gap i.e. $M(\mathbf{u}_{\text{worst},\mathbf{v}})$, let us assume the solution was such that $G_{\mathbf{u}_{\text{worst}}}$ only covers one mode of the true distribution (Figure 25 d)). Then $M(\mathbf{u}_{\text{worst},\mathbf{v}}) = \log(0.5) + \log(1-0.5)$. The minmax calculation is: $M(\mathbf{u}, \mathbf{v}_{\text{worst}}) = \log(0.5) + \log(1-0.5)$. Hence, the value of DG is zero, despite the collapse in the calculation for the $\mathbf{u}_{\text{worst}}$. The generator has no incentive to spread its mass due to the objective. While this is a problem for the original GAN that is being trained, it is not an issue for the calculation of the duality gap metric.

Figure 26 b) shows samples generated from $G_{\mathbf{u}_{\text{worst}}}$ when the experiment is performed in practice. We do observe that in this case, there is indeed a collapse that happened in the worst generator for



(a) GAN generator

(b) GAN discriminator

(c) DG: $\mathbf{v}_{\text{worst}}$

(d) DG: $\mathbf{u}_{\text{worst}}$

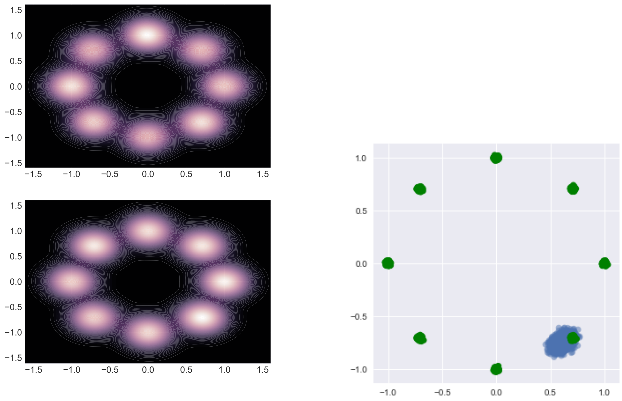
Figure 25: Analysis of a GAN that has reached the equilibrium for the mixture of 8 Gaussians problem. a) Samples generated from the GAN generator cover all 8 modes uniformly; b) Probabilities for a sample being real. The GAN discriminator assigns 0.5 probability to data points from the 8 modes and 0 everywhere else; c) For the computation of the duality gap, the theoretical $\mathbf{v}_{\text{worst}}$ assigns 0.5 to fake/real samples for the fixed GAN generator; d) We assume there was mode collapse when computing $\mathbf{u}_{\text{worst}}$ for the fixed GAN discriminator and samples from $G_{\mathbf{u}_{\text{worst}}}$ lie only on a single mode.

the fixed GAN discriminator. Yet, $D(G_{\mathbf{u}_{\text{worst}}}) = 0.489$ and $DG = 0.002$ confirming the previous thought experiment. A heatmap with generated samples from the GAN generator are given in Fig. 26.

Hence, mode collapse for the computation of DG is not an issue. Note though that when there is mode collapse in the GAN itself that is being evaluated, the DG detects this. In particular, this is supported by the high anti-correlation between DG and the number of covered modes and sample quality as shown in Table 7.

Suboptimal solutions due to the optimization. We now investigate the effect of the number of optimization steps used for the calculation of the duality gap on the quality of the solution. We run 5 different models with different hyperparameters with the goal to find the best setting. As suggested, we want to use the duality gap as the metric for this. Table 11 gives the results. The ranking of the models is the same for various numbers of optimization steps and corresponds to the ranking obtained by taking into consideration the number of covered modes and the number of generated samples that fall within 3 standard deviations of one of the modes.
This suggests that as long as one uses the same number of optimization steps when comparing different models, the suboptimality of the solution is empirically not an issue.

Comparison between the approximate DG and the true DG. We now investigate how the approximate DG compares to the true DG (which in this case is computed using an extensive grid search). Specifically, we compare the approximate DG (i) *DG-approx* to (ii) *DG-true-grid* and (iii) *DG-true-conv* (see Figure 27). The real data is a 1D Gaussian. Hence, for (ii) the true G_{worst} can be computed using an extensive grid search within a wide interval, whereas D_{worst} is computed by optimization till convergence. Similarly, for (iii) both D_{worst} and G_{worst} are optimized till convergence, whereas (i) uses only a few steps. We find that the gap between the true and approximate DG is not large. In particular, we detect strong correlation - (i) and (ii): 0.81, (i) and (iii): 0.89 (ii) and (iii): 0.92.



(a) GAN discriminator

(b) GAN generator

Figure 26: a) A heatmap of generated samples from the GAN generator (up) and the true data distribution (below). The generator is able to cover the true data distribution; b) Generated samples from $G_{\mathbf{u}_{\text{worst}}}$.

hyperparameters		quality of GAN		DG for # optimization steps			
lr_D	lr_G	#modes (out of 8)	# quality samples (out of 2500)	500	1000	1500	2000
2e-3	1e-4	8	2414	0.014	0.03	0.04	0.06
1e-4	1e-4	1	1119	10.02	11.3	12.23	12.3
1e-3	1e-4	8	2440	0.009	0.01	0.006	0.02
5e-3	1e-4	8	2478	0.008	0.002	0.002	0.001
1e-4	1e-5	1	501	10.84	12.37	13.25	13.7

Table 11: DG for various number of optimization steps and GAN hyperparameters. The set of the best hyperparameters is the same no matter the number of optimization steps are used for the calculation of the duality gap.

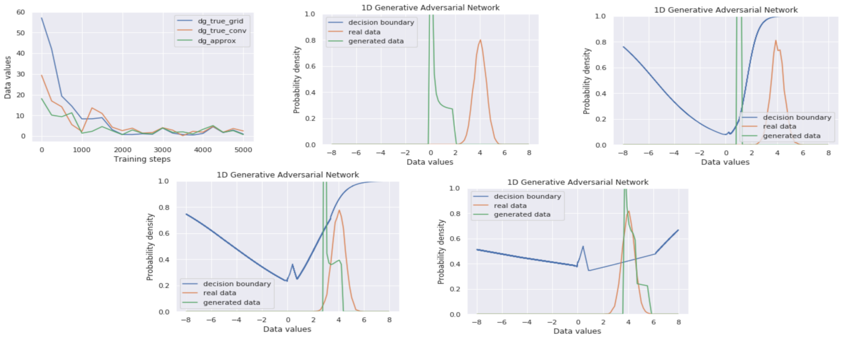


Figure 27: GAN trained on 1D Gaussian: (i) DG-approx vs. true DG; (i) beginning of training; (ii) after 500 steps; (iii) after 1500 steps; (iV) at the end of training