TOWARDS A TESTABLE NOTION OF GENERALIZATION FOR GENERATIVE ADVERSARIAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

We consider the question of how to assess generative adversarial networks, in particular with respect to whether or not they generalise beyond memorising the training data. We propose a simple procedure for assessing generative adversarial network performance based on a principled consideration of what the actual goal of generalisation is. Our approach involves using a test set to estimate the Wasserstein distance between the generative distribution produced by our procedure, and the underlying data distribution. We use this procedure to assess the performance of several modern generative adversarial network architectures. We find that this procedure is sensitive to the choice of ground metric on the underlying data space, and suggest a choice of ground metric that substantially improves performance. We finally suggest that attending to the ground metric used in Wasserstein generative adversarial network training may be fruitful, and provide a concrete formulation for doing so.

1 INTRODUCTION

Generative adversarial networks (GANs) (Goodfellow et al., 2014) have recently attracted significant interest as a means for performing unsupervised learning. However, recently concerns have been raised about their ability to generalise from training data and their capacity to overfit (Arora & Zhang, 2017; Arora et al., 2017). Moreover, techniques for evaluating the quality of GAN output are either ad hoc, lack theoretical rigor, or are not suitably objective – often times "visual inspection" of samples is the main tool of choice for practitioner. More fundamentally, it is sometimes unclear exactly what we want a GAN to do: what is the unsupervised learning task that we are trying to achieve?

In this paper, we provide a simple formulation of the GAN training framework, which consists of using a finite dataset to estimate an underlying true distribution. The quality of GAN output is measured quantitatively in terms of a statistical distance D between the estimated and true distribution. Within this context, we propose an intuitive notion of what it means for a GAN to generalise.

We also show how our notion of performance can be measured empirically for any given GAN, when D is chosen to be a Wasserstein distance. Unlike other methods (such as the inception score (Salimans et al., 2016)), we show that this choice of D requires us to make no density assumptions about the underlying true distribution. We investigate this choice of D empirically, showing that its performance is heavily dependent on the choice of ground metric underlying the Wasserstein distribution. We suggest a novel choice of ground metric that we show achieves good performance, and discuss how we might use this observation to improve the design of Wasserstein GANs (WGANs) (Arjovsky et al., 2017).

2 THE OBJECTIVE OF UNSUPERVISED LEARNING

In order to assess the effectiveness of GANs, it is essential that we define what we mean for a GAN to work well. We do so in this section by providing a simple but general formulation of the goal of unsupervised learning in this context. Our formulation is compatible with most of what is considered in the literature and what is done in practice, but we hope it will allow us to clarify certain details about the task.

To begin with, we assume some distribution π on a set $\mathcal{X}$. For instance, $\mathcal{X}$ may be the set of 32x32 colour images, and π the distribution from which the CIFAR-10 dataset was sampled. Informally, the goal will be to learn π in a particular sense.

We assume that π is almost completely intractable: we do not know its density (or even if it has one), and we do not have a procedure to draw new samples from it. However, we do suppose that we have a fixed dataset $X := \{x_1, \cdots, x_n\}$, where $x_i \overset{\text{iid}}{\sim} \pi$. Equivalently, we have the empirical distribution

$$\hat{X} := \frac{1}{n} \sum_{i=1}^{n} \delta_{x_i},$$

where δ denotes the Dirac mass.

Denote by $\mathcal{P}(\mathcal{X})$ the set of distributions on $\mathcal{X}$. We wish to use X to produce a distribution $\alpha(X) \in \mathcal{P}(\mathcal{X})$ that is as "close" as possible to π. We measure closeness measured in terms of a function $D : \mathcal{P}(\mathcal{X}) \times \mathcal{P}(\mathcal{X}) \to \mathbb{R}$. Usually D is chosen to be a *statistical divergence*, which means that $D(P, Q) \geq 0$ for all P and Q, with equality iff $P = Q$. Stated compactly, then, our goal as designers of unsupervised learning algorithms in this context is to choose α such that $D(\alpha(X), \pi)$ is usually small for the sorts of X and π we see in practice.

Now, we can immediately see that one possibility is simply to choose

$$\alpha(X) := \hat{X}.$$

Moreover, in the case that D is a metric for the weak topology on $\mathcal{P}(\mathcal{X})$, we have that

$$D(\hat{X}, \pi) \to 0$$

almost surely as $n \to \infty$, so that, assuming X is large enough, we can already expect $D(\hat{X}, \pi)$ to be small. This suggests a natural notion of *generalisation* in this context: we can say that a choice of α generalises for a given X if

$$D(\alpha(X), \pi) < D(\hat{X}, \pi). \tag{1}$$

In other words, using α here actually achieved something: in a sense, it has injected additional information about π into $\hat{X}$ (perhaps through some sort of smoothing or regularisation), and brought us closer to π than we already were *a priori*.

3 Generalisation in GANs

How do GANs fit into this framework? Most GAN algorithms in widespread use adhere to the following template. They take as input a distribution P, from which we assume we can sample, and compute (or approximate)

$$\Gamma(P) := \arg\min_{Q \in \mathcal{Q}} D_\Gamma(P, Q)$$

for some choices $\mathcal{Q} \subseteq \mathcal{P}(\mathcal{X})$ and $D_\Gamma : \mathcal{P}(\mathcal{X}) \times \mathcal{P}(\mathcal{X}) \to \mathbb{R}$. In other words, a GAN takes a distribution P from which only sampling is possible, and (in the ideal case) outputs the D_Γ-projection of P onto $\mathcal{Q}$.

In practice, $\mathcal{Q}$ denotes corresponds to the set of pushforward measures $G(\nu)$ obtained from a fixed noise distribution ν on a noise space $\mathcal{Z}$ and some set $\mathcal{G}$ of functions $G : \mathcal{Z} \to \mathcal{X}$. That is,

$$\mathcal{Q} = \{G(\nu) : G \in \mathcal{G}\}.$$

Usually G corresponds to the set of functions realisable by some neural network architecture, and ν is some multivariate uniform or Gaussian distribution.

For example, the original GAN formulation (Goodfellow et al., 2014) took D_Γ to be the Jenson-Shannon divergence, whereas the f-GAN (Nowozin et al., 2016) generalised this to arbitrary f-divergences, and the Wasserstein GAN (Arjovsky et al., 2017) advocated the Wasserstein distance. These works also made various recommendations about the choice $\mathcal{G}$ through various neural network architectural considerations. Many of the results proved in these papers involve showing that

(usually under some assumptions, such as sufficient discriminator capacity) a proposed objective for G is in fact equivalent to $D_\Gamma(P, G \circ \nu)$. For instance, the original GAN paper showed that

In terms of our framework in the previous section, then, using a GAN Γ amounts to choosing

$$\alpha(X) := \Gamma(\hat{X}) = \arg\min_{Q \in \mathcal{Q}} D_\Gamma(\hat{X}, Q).$$

It is a crucial point that, in most cases (e.g. (Radford et al., 2015)), Γ receives $\hat{X}$ as input rather than π itself: we only have access to finitely many CIFAR-10 samples, for example. Moreover, training GANs usually involves making many passes over the same dataset, so that, in effect, sampling from P returns many duplicate points over the run. We would not expect this to occur with nonzero probability if $P = \pi$ for most π of interest.

The observation that P is $\hat{X}$ rather than π itself was recently made by Arora et al. (2017). The authors argue that this introduces a problem for the ability of GANs to generalise, since, if D_Γ is a divergence (which is almost always the case), and if $\mathcal{Q}$ is too big (in particular, if it is big enough that $\hat{X} \in \mathcal{Q}$), then we have trivially that $\Gamma(\hat{X}) = \hat{X}$. In other words, the GAN objective appears to actively encourage $\Gamma(\hat{X})$ to memorise the dataset X, and never to produce novel samples from outside it.

The authors' proposed remedy involves trying to find a better choice of D_Γ. The problem, they argue, is that popular choices of D_Γ do not allow the approximation $D_\Gamma(\pi, Q) \approx D_\Gamma(\hat{X}, Q)$. They claim a good D_Γ is one that satisfies the condition (at least with high probability):

$$D_\Gamma(\pi, Q) \text{ is small whenever } D_\Gamma(\hat{X}, Q) \text{ is small.} \tag{2}$$

They point out that this is certainly violated when D_Γ is the Jensen-Shannon divergence, since

$$\mathrm{JS}(P \parallel Q) = \log 2$$

when one of P and Q is discrete and the other continuous. They give a similar result for the Wasserstein distance in the case that π is Gaussian. As a solution, they introduce the *neural network distance* D_{NN}, defined by

$$D_{\text{NN}}(P, Q) := \max_{f \in \mathcal{F}} \mathbb{E}_{x \sim P}\left[f(x)\right] - \mathbb{E}_{x \sim Q}\left[f(x)\right],$$

for some choice of a class of functions $\mathcal{F}$. They show that, assuming some smoothness conditions on the members of $\mathcal{F}$, the choice $D_\Gamma = D_{\text{NN}}$ satisfies (2) with high probability given a fairly modest sample size n.

However, (2) does not seem to capture an appealing notion of generalisation for GANs. Ultimately what matters is not that D_Γ generalises, but that $\Gamma(\hat{X})$ does. Indeed, fixating solely on (2) invites choosing D_Γ so as to bring Q artificially close to π when Q is close to $\hat{\pi}$ (with "closeness" measured in terms of d). This does not seem intuitively desirable; we see, for instance, that the degenerate choice $D_0(P, Q) := 0$ trivially satisfies (2). Moreover, we also note that, just like D_{NN}, D_0 is a pseudometric, and it is therefore unclear what mathematical properties of D_{NN} render it more suitable for estimating π than the obviously undesirable D_0. The authors do somewhat acknowledge this shortcoming of D_{NN}, pointing out that, for instance, $D_{\text{NN}}(P, Q)$ can be small even if P and Q are "quite different" (in some sense).

The issues raised in (Arora et al., 2017) apply only in the case that $\mathcal{Q}$ is too large. In practice, however, $\mathcal{Q}$ is heavily restricted (since $\mathcal{G}$ is restricted via a choice of neural network architecture), so that *a priori* we do not know whether $\Gamma(\hat{X}) = \hat{X}$ is even possible. As such, we do not see the choice of $\alpha(X) = \Gamma(\hat{X})$ necessarily as a bad idea. In fact, this choice falls perfectly within the framework of minimum distance estimation (MDE) (Wolfowitz, 1957; Basu et al., 2011), which involves estimating an underlying distribution by minimising a distance measure to a given empirical distribution.

We believe therefore that it is an open question as to how well GANs actually work. In particular, we propose to consider whether GANs work in the sense described in the previous section. In other words, what is the value of $D(\Gamma(\hat{X}), \pi)$? We turn to this question in the next section.

4 TESTING GANS

Our goal in this section is to compute $D(\alpha(X), \pi)$ for $\alpha(X) = \Gamma(\hat{X})$. This raises some difficulties, given that π is intractable. Our approach is to take D to be the first Wasserstein distance W_d defined by

$$W_d(P, Q) = \inf_{\gamma \in \Pi(P,Q)} \int_{\mathcal{X} \times \mathcal{X}} d(x, y) \, \mathrm{d}\gamma(x, y),$$

where d is a metric on $\mathcal{X}$ referred to as the *ground metric*, and $\Pi(P, Q)$ denotes the set of joint distributions on the product space $\mathcal{X} \times \mathcal{X}$ with marginals P and Q.

Before proceeding, we point out that it still makes sense to choose $D = W_d$ even if $D_\Gamma \neq W_d$. In our setup, D defines our end goal, whereas D_Γ simply defines a means to (possibly) achieve that goal, namely by allowing us to project onto $\mathcal{Q}$. If our goal is to minimise $W_d(\alpha(X), \pi)$, it is still conceivable that the original GAN formulation might be useful, even though in this case D_Γ is the Jenson-Shannon divergence – perhaps the choice of $\mathcal{Q}$ regularises $\hat{X}$ in such a way that brings us closer in W_d sense also.

Our choice of the Wasserstein distance means that D is sensitive to the topology of the underlying set $\mathcal{X}$, which we control by our choice of d. Moreover, W_d metricizes weak convergence for the Wasserstein space $\mathcal{P}_d(\mathcal{X})$ defined by

$$\mathcal{P}_d(\mathcal{X}) := \left\{ P \in \mathcal{P}(\mathcal{X}) : \int_{\mathcal{X}} d(x_0, y) \, \mathrm{d}P(y) < \infty \text{ for some } x_0 \in \mathcal{X} \right\}$$

(Villani, 2008). Consequently, if we denote by A a set of samples $a_1, \cdots, a_m \overset{\text{iid}}{\sim} \alpha(X)$, and by Y a set of samples (separate from X) $y_1, \cdots, y_k \overset{\text{iid}}{\sim} \pi$, with $\hat{A}$ and $\hat{Y}$ the corresponding empirical distributions, then, provided

$$\alpha(X), \pi \in \mathcal{P}_d(\mathcal{X}), \tag{3}$$

we have that $D(\hat{A}, \hat{Y}) \to D(\alpha(X), \pi)$ almost surely as $\min\{m, k\} \to \infty$. Note that condition (3) holds automatically in the case that $(\mathcal{X}, d)$ is compact, since then $\mathcal{P}_d(\mathcal{X}) = \mathcal{P}(\mathcal{X})$.

As such, to estimate $D(\alpha(X), \pi)$, for $D = W_d$, we propose the following. Before training, we divide all the samples from π that we have into a training set X and a testing set Y. We train our GAN on X, obtaining $\alpha(X)$. We then take samples A from $\alpha(X)$, and obtain an estimate of $D(\alpha(X), \pi)$ by computing $D(\hat{A}, \hat{Y})$, which can be done exactly by solving a linear program (Villani, 2003). We can also use the same methodology to estimate $D(\hat{X}, \pi)$ by $D(\hat{X}, \hat{Y})$, which allows us to check whether 1 holds by checking whether

$$D(\hat{A}, \hat{Y}) < D(\hat{X}, \hat{Y}). \tag{4}$$

4.1 RESULTS

We used our methodology to test two popular GAN varieties – the DCGAN (Radford et al., 2015) and the Improved Wasserstein GAN (I-WGAN) (Gulrajani et al., 2017) – on the MNIST and CIFAR-10 datasets. In all cases, A and Y consisted of 10000 samples.

4.1.1 L^2 AS GROUND METRIC

Initially, we took our ground metric d to be the L^2 distance, which has the appealing property of making $(\mathcal{X}, d)$ compact when $\mathcal{X}$ is a space of RGB or greyscale images, therefore ensuring that (3) holds. We found that this choice produced nice results for the I-WGAN. Figure 1 shows the value of $W_{L^2}(\hat{G}, \hat{Y})$ over multiple training runs for MNIST and CIFAR-10. In both cases, this curve decays towards an asymptote that nicely corresponds to the visual quality of the samples produced, examples of which are shown in Figures 6 and 7 in the appendix. Moreover, we see that $W_{L^2}(\hat{G}, \hat{Y})$ is much closer to $W_{L^2}(\hat{X}, \hat{Y})$ towards the tail end of training for MNIST than for CIFAR-10. This seems to reflect the fact that, visually, the eventual MNIST samples produced by the I-WGAN do seem to match more closely the true MNIST distribution than the I-WGAN CIFAR-10 samples, which are easily identified as "fake".

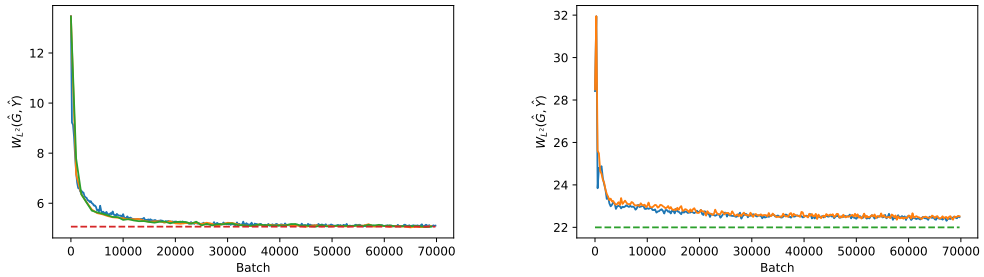


Figure 1: $W_{L^2}(\hat{G}, \hat{Y})$ over training run for I-WGAN trained on MNIST (left) and CIFAR-10 (right). $W_{L^2}(\hat{X}, \hat{Y})$ is shown dashed in both cases.

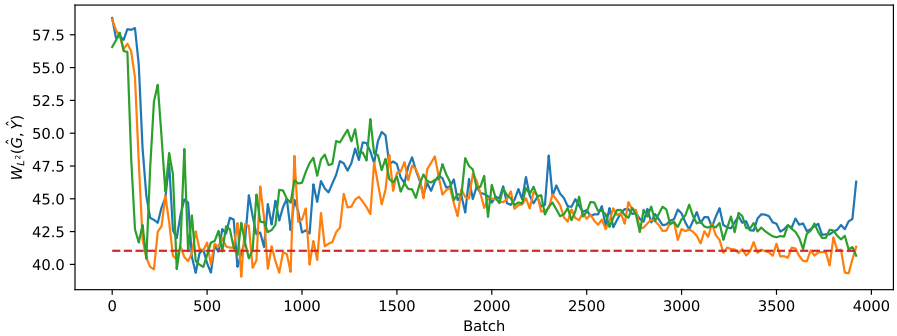


Figure 2: $W_{L^2}(\hat{G}, \hat{Y})$ over training run for DCGAN trained on MNIST. $W_{L^2}(\hat{X}, \hat{Y})$ is shown dashed.

However, we did not obtain such a nice result when we ran the same experiment on CIFAR-10 samples produced by a DCGAN. In fact, the $W_{L^2}(\hat{G}, \hat{Y})$ curves produced, shown in Figure 2 apparently show that $\hat{G}$ is *closer* to $\hat{Y}$ than $\hat{X}$ very early on in training, at around batch 500. As observed in Figure 8 in the appendix, the visual quality of samples at this stage is very poor. This is both intriguing and concerning: our metric considers 10000 samples whose appearance resembles the heavily blurred Figure 8b to be closer to $\hat{Y}$ than 10000 *actual* π samples sourced from X.

Motivated by the visual blurriness of the samples in Figure 8b, we explored the effect on $W_{L^p}(\hat{X}, \hat{Y})$ of blurring the CIFAR-10 training set X. In particular, we let X and Y each consist of 10000 distinct CIFAR-10 samples in X. We then independently convolved each channel of each image with a Gaussian kernel having standard deviation σ, obtaining a blurred dataset $\beta_\sigma(X)$ and corresponding empirical distribution $\beta_\sigma(\hat{X})$. The visual effect of this procedure is shown in Figure 9 in the appendix. We then computed $W_{L^p}(\beta_\sigma(\hat{X}), \hat{Y})$ with σ ranging between 0 and 10 for a variety of values of p. The results of this experiment in the case $p = 2$ are shown in Figure 3, and similar results were observed for other values of p: in all cases, we found that

$$W_{L^p}(\hat{X}, \hat{Y}) < W_{L^p}(\beta_\sigma(\hat{X}), \hat{Y})$$

whenever $\sigma > 0$. That is, blurring X by any amount brings $\hat{X}$ closer to $\hat{Y}$ in W_{L^p} than not. This occurs even though X is distributed identically to Y (both being drawn from π), while $\beta_\sigma(X)$ (presumably) is not when $\sigma > 0$.

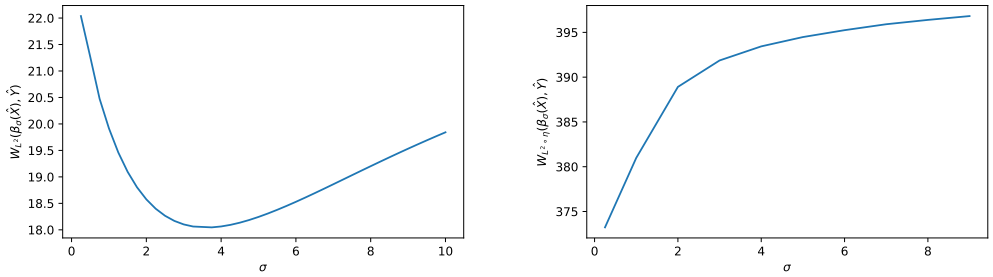


Figure 3: Effect of σ on $W_d(\beta_\sigma(\hat{X}), \hat{Y})$ with X and Y obtained from CIFAR-10, for $d = L^2$ (left) and $d = L^2 \circ \eta$ (right)

4.1.2 EMBEDDED L^2 AS GROUND METRIC

To remedy these issues, we sought to replace L^2 with a choice of d that is more naturally suited to the space of images in question. Our idea involves mapping $\mathcal{X}$ through a fixed pre-trained neural network η into a feature space $\mathcal{Y}$, and then computing distances using some metric $d_\mathcal{Y}$ on $\mathcal{Y}$, rather than in $\mathcal{X}$ directly. It is easily seen that, provided η is injective, $d_\mathcal{Y} \circ \eta : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ defined by

$$d_\mathcal{Y} \circ \eta(x, x') := d_\mathcal{Y}(\eta(x), \eta(x'))$$

is a metric. It also holds that, when η is $(d, d_\mathcal{Y})$-continuous, $(\mathcal{X}, d_\mathcal{Y} \circ \eta)$ is compact when $(\mathcal{X}, d)$ is: given a sequence $x_i \subseteq \mathcal{X}$, there exists a subsequence $x_{i'}$ that converges in d to some x (by compactness), so that $d_\mathcal{Y} \circ \eta(x_{i'}, x) = d_\mathcal{Y}(\eta(x_{i'}), \eta(x)) \to 0$ by continuity of η. Neither of these conditions on η – injectivity and $(d, d_\mathcal{Y})$-continuity – are unreasonable to expect from a typical neural network trained using stochastic gradient descent (at least, when $d_\mathcal{Y}$ is a typical metric such as an L^p distance). Consequently, $W_{d_\mathcal{Y} \circ \eta}$ constitutes a valid metric on $\mathcal{P}_{d_\mathcal{Y} \circ \eta}(\mathcal{X})$, and (3) is automatically satisfied.

To test the performance of $W_{d_\mathcal{Y} \circ \eta}$, we repeated the blurring experiment described above. We took $\eta(x)$ to be the result of scaling $x \in \mathcal{X}$ to size 224x224, mapping the result through a DenseNet-121 (Huang et al., 2016) pre-trained on ImageNet (Deng et al., 2009), and extracting features immediately before the linear output layer. Under the same experimental setup as above otherwise, we obtained the plot of $W_{L^2 \circ \eta}(\beta_\sigma(\hat{X}), \hat{Y})$ shown in Figure 3. Happily, we now see that this curve increases monotonically as σ grows in accordance with the declining visual quality of $\beta_\sigma(X)$ shown in Figure 9.

Next, we reran our earlier experiments, this time computing $W_{L^2 \circ \eta}(\hat{A}, \hat{Y})$ over the course of training. In all cases, we observed that the value of this quantity reflected very well the visual quality of the training samples. For the I-WGAN, we used the same η as above, and obtained the results on MNIST and CIFAR-10 shown in Figure 4. (Note that on MNIST we modified first duplicated each input across three channels for shape compatibility with the DenseNet.) For the DCGAN on CIFAR-10, we obtained the curve shown in Figure 5.

In all cases we observe that $W_{L^2 \circ \eta}(\hat{A}, \hat{Y})$ decreases monotonically towards an asymptote in a way that accurately summarises the visual quality of the samples throughout the training run. Moreover, in all cases there is a large gap between the eventual value of $W_{L^2 \circ \eta}(\hat{A}, \hat{Y})$ and $W_{L^2 \circ \eta}(\hat{X}, \hat{Y})$, which reflects the fact that the GAN samples produced are ultimately distinguishable from real samples from π. In particular, we see an improvement in this respect for the I-WGAN on MNIST: previously, the final value of $W_{L^2}(\hat{A}, \hat{Y})$ was essentially the same as $W_{L^2}(\hat{X}, \hat{Y})$, despite the fact that it is still quite easy to distinguish between real and generated samples (see e.g. the various mistakes that are discernible throughout Figure 6)

5 DISCUSSION AND FUTURE WORK

We believe our work reveals two promising avenues of future inquiry. First, we suggest that $W_{L^p \circ \eta}$ is an appealing choice of D, both due to its nice theoretical properties – it metricizing weak con-

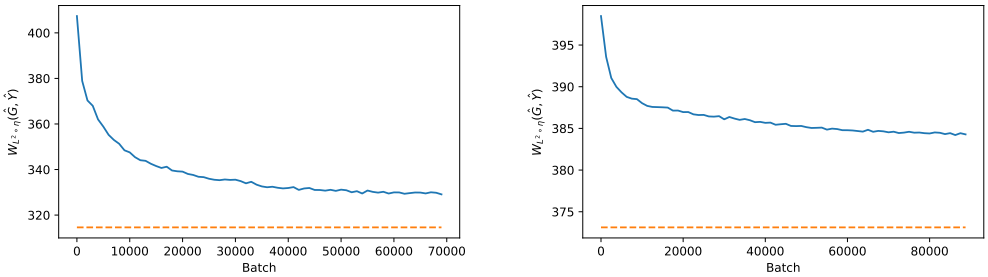


Figure 4: $W_{L^2 \circ \eta}(\hat{G}, \hat{Y})$ over training run for I-WGAN trained on MNIST (left) and CIFAR-10 (right). $W_{L^2 \circ \eta}(\hat{X}, \hat{Y})$ is shown dashed in both cases.

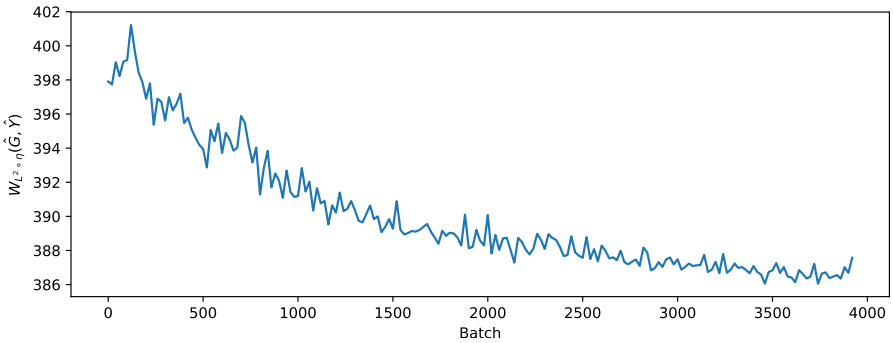


Figure 5: $W_{L^2 \circ \eta}(\hat{G}, \hat{Y})$ over training run for DCGAN trained on MNIST. $W_{L^2 \circ \eta}(\hat{X}, \hat{Y})$ is shown dashed.

vergence, and does not require us to make any density assumptions about π – and due to its sound empirical performance demonstrated above. It would be very interesting to use this D to produce a systematic and objective comparison of the performance of all current major GAN implementations, and indeed to use this as a metric for assessing future GAN designs. We also view the test (4) as potentially useful for determining whether our algorithms are overfitting. This would be particularly so if applied via a cross-validation procedure: if we consistently observe that (4) holds when training a GAN according to many different X and Y partitions of our total π samples, then it seems reasonable to infer that $\alpha(X)$ has indeed learnt something useful about π.

We also believe that the empirical inadequacy of W_{L^2} that we observed suggests a path towards a better WGAN architecture. At present, WGAN implementations implicitly use W_{L^2} for their choice of D_Γ. We suspect that altering this to our suggested $W_{L^2 \circ \eta}$ may yield better quality samples. We briefly give here one possible way to do so that is largely compatible with existing WGAN setups. In particular, following Arjovsky et al. (2017), we take

$$D_\Gamma(P, Q) = \max_{f \in \mathcal{F}} \mathbb{E}_{x \sim P}\left[f(x)\right] - \mathbb{E}_{x \sim Q}\left[f(x)\right]$$

for a class $\mathcal{F}$ of functions $f : \mathcal{X} \to \mathbb{R}$ that are all $(L^2 \circ \eta, d_\mathbb{R})$-Lipschitz for some fixed Lipschitz constant K. Here $d_\mathbb{R}$ denotes the usual distance on $\mathbb{R}$. To construct such an $\mathcal{F}$ in practice, we can simply consider a space $\mathcal{F}'$ of $(d_\mathcal{Y}, d_\mathbb{R})$-Lipschitz functions with fixed Lipschitz constant L, which may be done for instance via weight-clipping like Arjovsky et al. (2017), or via a gradient penalty like Gulrajani et al. (2017). Provided η is itself $(d, d_\mathcal{Y})$-Lipschitz with some Lipschitz constant M (which, for a neural network with fixed weights, can easily be checked), we then obtain a KM-Lipschitz function $f : \mathcal{X} \to \mathbb{R}$ by taking

$$f(x) := h(\eta(x)).$$

Moreover, for our suggested η, this still enables us to compute the gradient of f via the chain rule, since η is a neural network and therefore assumed to be differentiable. We have begun some experimentation in this area but leave a more detailed empirical inquiry to future work.

It is also clearly important to establish better theoretical guarantees for our method. At present, we have no guarantee that the number of samples in A and Y are enough to ensure that

$$D(\hat{A}, \hat{Y}) \approx D(\alpha(X), \hat{Y})$$

(perhaps with some fixed bias that is fairly independent of α, so that it is valid to use the value of $D(\hat{A}, \hat{Y})$ to compare different choices of α), or that (4) entails (1) with high probability. We do however note that some recent theoretical work on the convergence rate of empirical Wasserstein estimations (Weed & Bach, 2017) does suggest that it is plausible to hope for fast convergence of $D(\hat{A}, \hat{Y})$ to $D(\alpha(X), \hat{Y})$. We also believe that the convincing empirical behaviour of $W_{L^2 \circ \eta}$ does suggest that it is possible to say something more substantial about our approach, which we leave to future work.

6 Related Work

The maximum mean discrepancy (MMD) is another well-known notion of distance on probability distributions, which has been used for testing whether two distributions are the same or not (Gretton et al., 2012) and alo for learning generative models in the style of GAN (Li et al., 2015; Dziugaite et al., 2015; Sutherland et al., 2016; Li et al., 2017). It is parameterized by a characteristic kernel k, and defines the distance between probability distributions by means of the distance of k's reproducing kernel Hilbert space (RKHS). The MMD induces the same weak topology on distributions as the one of the Wasserstein distance. Under a mild condition, the MMD between distributions P and Q under a kernel k can be understood as the outcome of the following two-step calculation. First, we push-forward P and Q from their original space $\mathcal{X}$ to a Hilbert space $\mathcal{H}$ (isomorphic to k's RKHS) using a feature function $\phi : \mathcal{X} \to \mathcal{H}$ induced by the kernel k. Typically, $\mathcal{H}$ is an *infinite*-dimensional space, such as the set ℓ_2 of sqaure-summable sequences in $\mathcal{R}^\infty$ as in Mercer's theorem. Let P' and Q' be the resulting distributions on the feature space. Second, we compute the supremum of $\mathbb{E}_{P'(X)}[f(X)] - \mathbb{E}_{Q'(X)}[f(X)]$ over all *linear* 1-Lipschitz functions $f : \mathcal{H} \to \mathbb{R}$. The result of this calculation is the MMD between P and Q.

This two-step calculation shows the key difference between MMD and our use of Wasserstein distance and neural embedding η. While the co-domain of the feature function ϕ is an *infinite*-dimensional space (e.g. ℓ_2) in most cases, its counterpart η in our setting uses a *finite*-dimensional space as co-domain. This means that the MMD possibly uses a richer feature space than our approach. On the other hand, the MMD takes the supremum over only *linear* functions f among all 1-Lipschitz functions, whereas the Wasserstein distance considers all these 1-Lipschitz functions. These difference balancing acts between the expressiveness of features and that of functions taken in the supremum affect the learning and testing of various GAN approaches as observed experimentally in the literature. One interesting future direction is to carry out systematic study on the theoretical and practical implications of these differences.

References

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

Sanjeev Arora and Yi Zhang. Do gans actually learn the distribution? an empirical study. *arXiv preprint arXiv:1706.08224*, 2017.

Sanjeev Arora, Rong Ge, Yingyu Liang, Tengyu Ma, and Yi Zhang. Generalization and Equilibrium in Generative Adversarial Nets (GANs). *arXiv preprint arXiv:1703.00573*, 2017.

Ayanendranath Basu, Hiroyuki Shioya, and Chanseok Park. *Statistical inference: the minimum distance approach*. CRC Press, 2011.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pp. 248–255. IEEE, 2009.

Gintare Karolina Dziugaite, Daniel M. Roy, and Zoubin Ghahramani. Training generative neural networks via maximum mean discrepancy optimization. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence, UAI 2015, July 12-16, 2015, Amsterdam, The Netherlands*, pp. 258–267, 2015.

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*, pp. 2672–2680, 2014.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander J. Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13:723–773, 2012.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron Courville. Improved training of wasserstein gans. *arXiv preprint arXiv:1704.00028*, 2017.

Gao Huang, Zhuang Liu, Kilian Q Weinberger, and Laurens van der Maaten. Densely connected convolutional networks. *arXiv preprint arXiv:1608.06993*, 2016.

Chun-Liang Li, Wei-Cheng Chang, Yu Cheng, Yiming Yang, and Barnabás Póczos. MMD GAN: towards deeper understanding of moment matching network. *arXiv preprint arXiv:1705.08584*, 2017.

Yujia Li, Kevin Swersky, and Richard S. Zemel. Generative moment matching networks. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pp. 1718–1727, 2015.

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

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

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*, pp. 2234–2242, 2016.

Dougal J. Sutherland, Hsiao-Yu Fish Tung, Heiko Strathmann, Soumyajit De, Aaditya Ramdas, Alexander J. Smola, and Arthur Gretton. Generative models and model criticism via optimized maximum mean discrepancy. *arXiv preprint arXiv:1611.04488*, 2016.

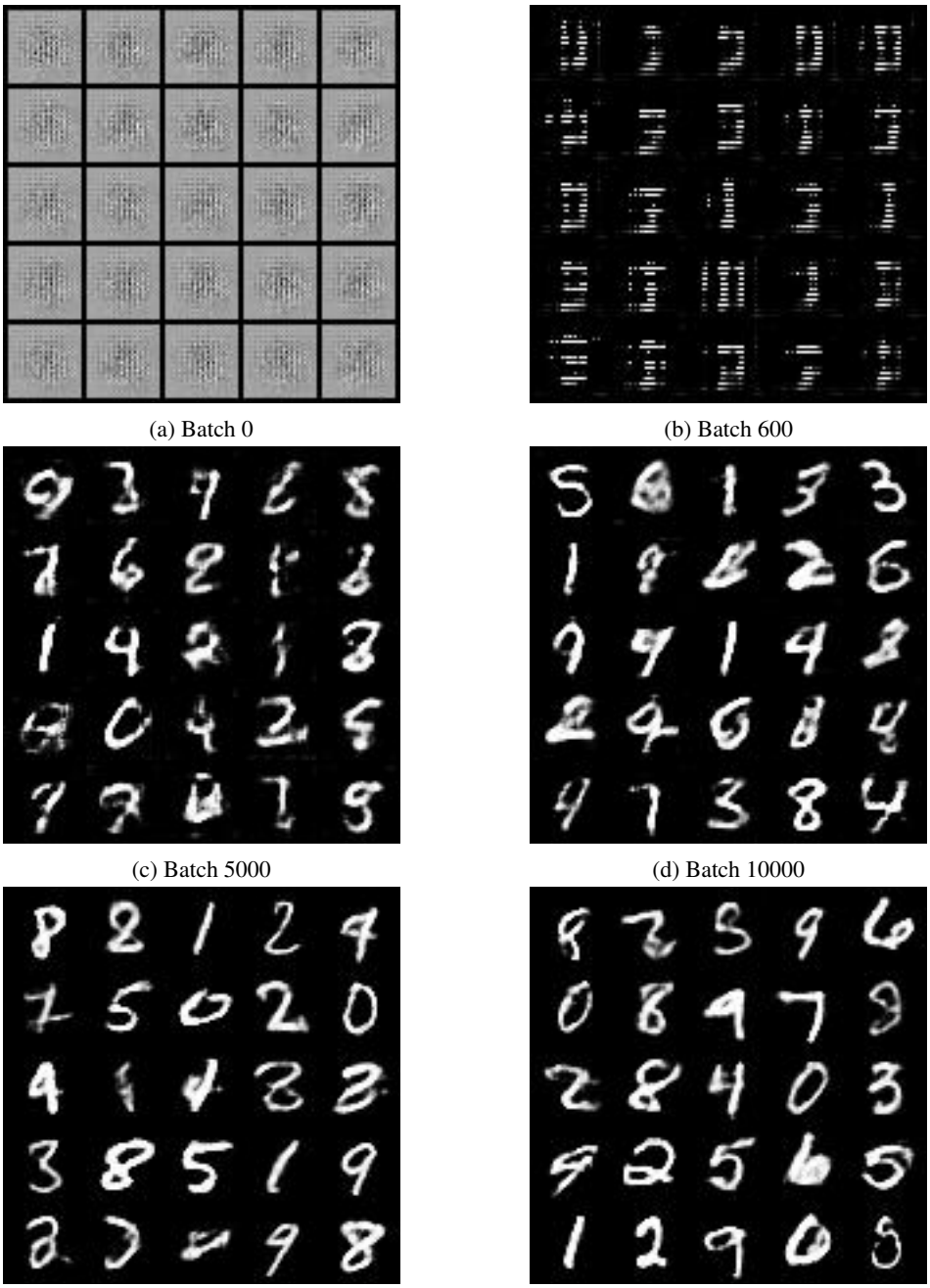
Cédric Villani. *Topics in optimal transportation*. Number 58. American Mathematical Soc., 2003.

Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

Jonathan Weed and Francis Bach. Sharp asymptotic and finite-sample rates of convergence of empirical measures in wasserstein distance. *arXiv preprint arXiv:1707.00087*, 2017.

Jacob Wolfowitz. The minimum distance method. *The Annals of Mathematical Statistics*, pp. 75–88, 1957.

A EXAMPLE I-WGAN SAMPLES

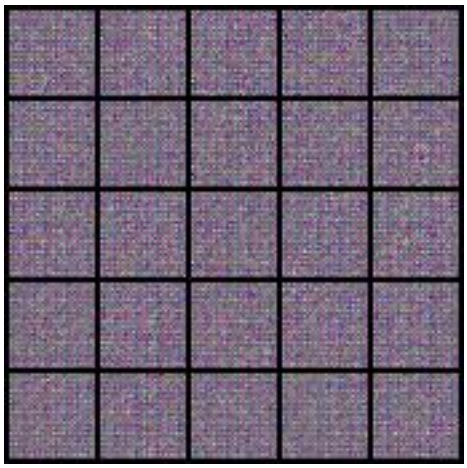


(a) Batch 0 (b) Batch 600

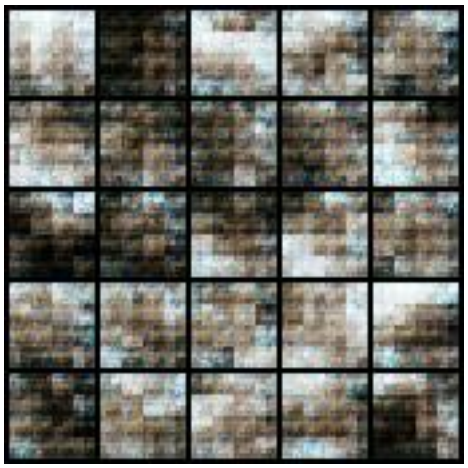
(c) Batch 5000 (d) Batch 10000

(e) Batch 50000 (f) Batch 69800

Figure 6: Samples from I-WGAN trained on MNIST



(a) Batch 0



(b) Batch 500

(c) Batch 5000

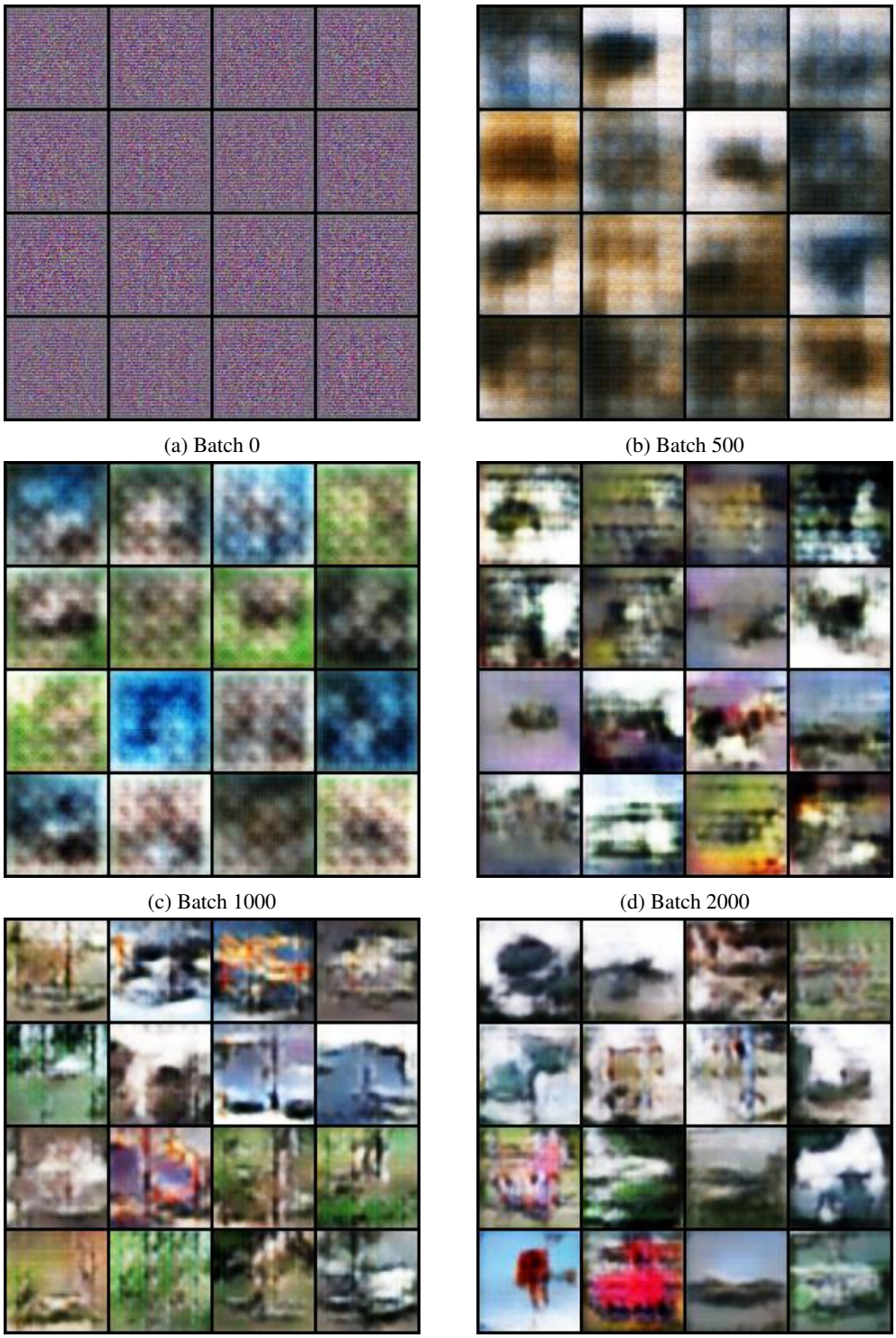
(d) Batch 10000

(e) Batch 50000

(f) Batch 69750

Figure 7: Samples from I-WGAN trained on CIFAR-10

B EXAMPLE DCGAN SAMPLES



(a) Batch 0 (b) Batch 500

(c) Batch 1000 (d) Batch 2000

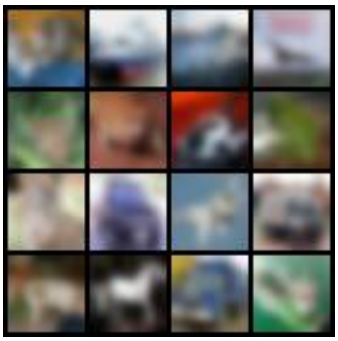
(e) Batch 3000 (f) Batch 3920

Figure 8: Samples from I-WGAN trained on CIFAR-10 test set

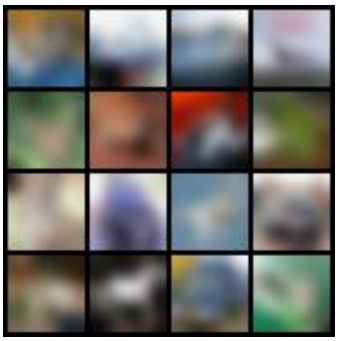
C EFFECT OF BLURRING CIFAR-10

(a) Unblurred

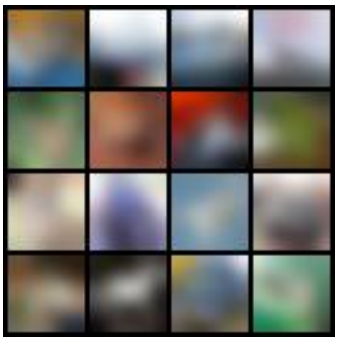
(b) $\sigma = 1$



(c) $\sigma = 2$



(d) $\sigma = 3$



(e) $\sigma = 4$

(f) $\sigma = 6$

(g) $\sigma = 8$

(h) $\sigma = 10$

Figure 9: Effect of different σ on $\beta_\sigma(X)$, for X the CIFAR-10 training set